TEXTURE CLASSIFICATION AND SEGMENTATION USING ANGULAR WAVELET FRAMES

By

K. Murali Mohan

DEPARTMENT OF ELECTRICAL ENGINEERING

Indian Institute of Technology Kanpur

APRIL, 2002

TEXTURE CLASSIFICATION AND SEGMENTATION USING ANGULAR WAVELET FRAMES

A *Thesis* Submitted

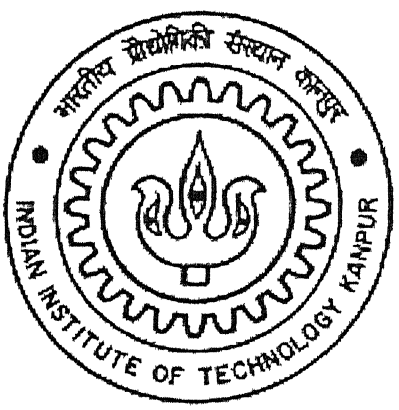
In Partial Fulfillment of the Requirements

For the Degree of

Master of Technology

By

K. Murali Mohan

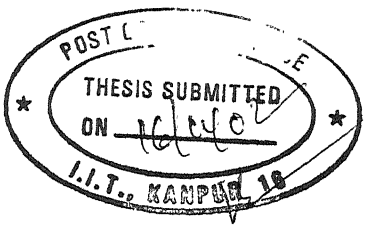


To the

DEPARTMENT OF ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY, KANPUR

April 2002



CERTIFICATE

This is to certify that the thesis work entitled **TEXTURE CLASSIFICATION AND SEGMENTATION USING ANGULAR WAVELET FRAMES** by *K. Murali Mohan*, Roll No. Y010416 has been carried out under my supervision and the same has not been submitted elsewhere for a degree.

Thesis Supervisor,

Dr. Sumana Gupta,

Professor, Dept. of Electrical Engineering,

Indian Institute of Technology, Kanpur.

April 2002

Dedicated

To

Sai Ram

ACKNOWLEDGEMENTS

I would like to thank my thesis guide **Dr. Sumana Gupta** for proposing the thesis problem and would like to express my sincere gratitude for her kind supervision, constant guidance, and encouragement through out my thesis work. She always guided me with her valuable ideas whenever I approached her. She guided me in reading the relevant literature and gave me important ideas and suggestions to mould the thesis work in a meaningful shape and immensely helped me in all respects.

I am grateful to EE department, IIT, Kanpur for providing me the opportunity and necessary help in pursuing my Masters program.

ABSTRACT

In this thesis we describe a new approach to characterize texture images at multiple scales using angular wavelet frames (AWF). We have constructed 2D AWF by applying frequency transformations to the prototype filters of a 1D perfect reconstruction filter bank (PRFB) to obtain 2D filters. The objective of constructing AWF is to improve the performance of existing wavelet based methods for classification and segmentation of textures having dominant angular components. A texture is characterized by a set of channel variances estimated at the output of the corresponding 2D filter bank. Classification results for 26 Brodatz textures indicate that performance of the proposed AWF method is comparable to the existing wavelet based algorithms. The constructed AWF is also applied to perform supervised texture segmentation. We considered several images with different number of texture regions, and used the standard k - means clustering algorithm to integrate the feature images and produce segmentation of the original texture images. The performance of the supervised texture segmentation algorithm using AWF is found satisfactory.

Contents

List of Figures

List of Tables

Chapter 1

INTRODUCTION

1.1. An Overview

Texture can be used in the analysis of images in several ways: in the segmentation of scenes into distinct objects and regions, in the classification or recognition of surface materials, and in the computation of surface shape. A perceptual surface texture may be informally defined to be a spatial pattern of local surface radiances giving rise to a perception of surface homogeneity. Within this context, an image texture may be defined as a local arrangement of image irradiances projected from a surface patch of perceptually homogeneous radiances.

Textures provide important characteristics for surface and object identification from aerial or satellite photographs, biomedical images, and many other types of images. Their analysis is fundamental to many applications such as industrial monitoring of product quality, remote sensing of earth resources, and medical diagnosis. The analysis of texture requires the identification of those texture attributes, which can be used for segmentation, discrimination, recognition, or shape computation and the development of computational approaches for accomplishing these tasks. Texture segmentation involves partitioning a digital image into a

finite set of regions with uniform textures. This problem plays an important role in the field of image processing and computer vision. To design an effective algorithm for texture segmentation, it is essential to find a set of texture features with good discriminating power.

1.2. Existing Methods

There exists a variety of feature extraction methods in texture analysis such as gray level co-occurrence matrix (GLCM) [1], Laws' texture energy (LAWS), second – order statistics, Gauss – Markov random field models, local linear transforms etc. They are restricted to the analysis of relatively small neighborhoods. As a consequence, their performance is best for the analysis and segmentation of the class of so-called micro textures. Recent models of human and mammalian vision suggest the existence of an internal spatial/frequency representation that is capable of preserving both local and global information. These findings have been the basis for several approaches to texture using bank of Gabor filters with different scale and orientation tuning [5], [6], [8], [9], [12].

The Gabor channel filters are complex sinusoidal (carrier) gratings modulated by 2D Gaussian functions in the space domain, and shifted Gaussians in the spatial frequency domain. The Gabor functions are complex valued functions on R^2, except when the carrier sinusoid has zero center frequency. In this case, the Gabor functions reduce to the 2D Gaussian functions. Gabor channel filters are well localized in space and frequency. This technique extracts features by filtering the texture image with a selected subset of a fixed Gabor filter bank. Several different criterions have been developed for the selection of the "right" subset of the filter bank, such as the filtered image energy [5] or the local maxima in the Fourier transform of the texture image [6]. The potential disadvantage of decompositions using the Gabor filter bank is that they are computationally quite intensive, especially for the

evaluation of low-frequency components. In addition, the outputs of Gabor filter banks are not mutually orthogonal, which may result in a significant correlation between texture features. Finally, if the center frequency of a selected Gabor filter does not match any of the important harmonics in the texture image it produces only "noisy" information and therefore, it degrades the segmentation performance. Most of these problems can be avoided by using wavelet transform.

The wavelet transform methods provide a precise and unifying framework for the analysis and characterization of a signal at different scales [2]-[4]. The use of a pyramid-structured wavelet transform for texture analysis was first suggested by Mallat [3]. This initial proposal has been followed by several studies on texture classification with a particular attention to the use of wavelet packets [7]. The algorithm developed by Chang and Kuo yields a texture specific wavelet tree stricture, which is used as a feature for classification. Unfortunately, this algorithm is not suitable for texture image segmentation because the extracted features are based on the wavelet tree structure of the entire image. However, each pixel of the images has to be classified and thus a feature vector for each pixel is needed.

Unser [10] proposed an overcomplete wavelet decomposition, which he called Discrete Wavelet Frame (DWF), by incorporating redundant information to yield a translation invariant description of textures. The DWF feature extraction technique was used for texture discrimination and can also be incorporated into a multidimensional texture segmentation algorithm. These wavelet-based methods decompose the frequency plane into four rectangular segments. As a result, their analysis of textures with dominant angular component is rather poor.

His-chin Hsin [13] proposed modulated wavelet transform with a modulating frequency pair, which decomposes the image in the frequency region centered at the modulating frequency pair. The modulated wavelet transform zooms in the frequency region centered at the modulating frequencies for further decompositions. However, the selection of suitable modulating frequencies is a difficult task.

1.3. Objective of Thesis

Gabor channel filters are well localized in both space and frequency but they are computationally quite intensive. Also the outputs of the Gabor channels are not mutually orthogonal, which result in correlation between texture features. The wavelet-based methods cannot capture the dominant angular frequency information. To over come these problems, we propose the construction of angular wavelet frames (AWF). To capture dominant angular frequency information, we construct 2D angular wavelet frames using circular and angular transformations on 1D prototype filter. The circular transformation on 1D low pass filter gives a circularly symmetric low pass filter which has non zero value for $\omega_1^2+\omega_2^2 < \pi/2$ and is zero for $\omega_1^2+\omega_2^2 > \pi/2$. The angular transformation on 1D low pass filter gives a 2D filter, which selects an angular region between 0 and $\pi/2$ in 2D positive frequency plane. We construct the 2D angular wavelet frames by using the combination of 2D circular and angular transformed filters. We also examine its performance for texture classification and segmentation.

1.4. Organization of Thesis

In chapter 1, a brief review of the existing methods used for the classification and segmentation of texture images is presented.

In chapter 2, we give the definition of the filter banks and discuss the main properties of the orthogonal wavelet transform in l_2 (the space of square summable sequences). This approach is further extended to the case of an over complete representation, and the concept of an orthogonal (or tight) wavelet frame is introduced. Finally we explain the method of constructing 2D angular wavelet frames using frequency transformations.

In chapter 3, texture classification and segmentation using 2D Angular Wavelet Frames is presented. Chapter 4 gives results obtained for texture classification and segmentation using Angular Wavelet Frames.

Chapter 2

WAVELET TRANSFORMS AND FRAMES

The wavelet transform is usually described as multiresolution decomposition for finite energy functions f of the continuous variable x (i.e., $f(x) \in L_2$) [3]. In the present context, it is more appropriate to consider wavelet representations for discrete signals in l_2 (the space of square summable sequences), which was initially proposed by Rioul [14]. Section 2.1 gives the definition of the underlying filter banks and 2.2 provides a brief review of the main properties of the orthogonal wavelet transform in l_2. In Section 2.3, this approach is extended to the case of an over complete representation, and the concept of an orthogonal (or tight) wavelet frame is introduced. In Section 2.4, 2D Angular Wavelet Frame design using frequency transformation is presented.

2.1. Filter Bank Specification and Properties

In this section, we characterize the perfect reconstruction filterbanks (PRFB) underlying the definition of the discrete wavelet transform [14], that we have used for our work. We start with the prototype filter h which satisfy the condition

$$H(z)\widetilde{H}(z)+G(z)\widetilde{G}(z)=1 \tag{2.1}$$

where H(z) is the z – transform of the low-pass filter h and g is its complementary high pass filter and is given by

$$G(z)=zH(-z^{-1}) \tag{2.2}$$

$\widetilde{H}(z)$ and $\widetilde{G}(z)$ are orthogonal or biorthogonal complements of filters H(z) and G(z) respectively. These prototype filters are then used to generate, in an iterative fashion, a sequence of filters of increasing width (indexed by i)

$$\begin{aligned} H_{i+1}(z) &= H(z^{2^i})H_i(z) \\ G_{i+1}(z) &= G(z^{2^i})H_i(z), \quad (i=0,1,...,I-1) \end{aligned} \tag{2.3}$$

with the initial condition $H_0(z) = 1$.

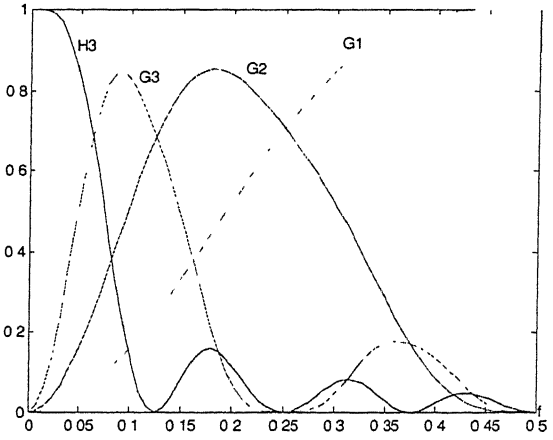
Equivalently, in the signal domain, we have the two – scale relation

$$\begin{aligned} h_{i+1}(k) &= [h]_{\uparrow 2^i} * h_i(k) \\ g_{i+1}(k) &= [g]_{\uparrow 2^i} * h_i(k) \end{aligned} \tag{2.4}$$

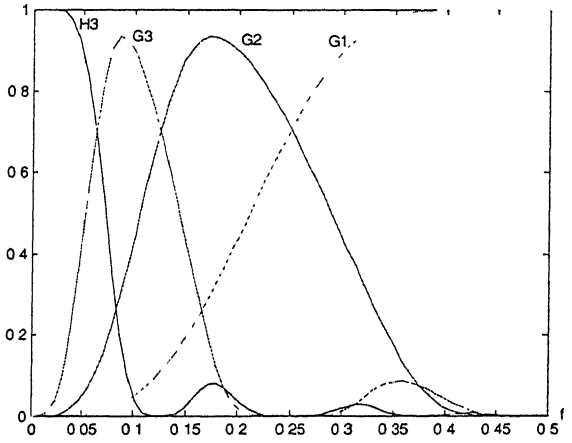
where the notation $[\cdot]_{\uparrow m}$ denotes the upsampling by a factor of m. The effect of one iteration is more or less to dilate the filters h_i and g_i by a factor of two. Such a sequence of filters can be used to decompose a signal in subbands of approximately one octave each. The properties of such decomposition are best understood by analyzing the multi-band characteristics of the underlying filter bank. Some examples are shown in Fig. 1. It can easily be shown that such sequences of filters also satisfy the identity

$$H_i(e^{j\omega})\widetilde{H}_i(e^{j\omega})+\sum_{k=1}^{i}G_k(e^{j\omega})\widetilde{G}_k(e^{j\omega}) = 1 \tag{2.5}$$

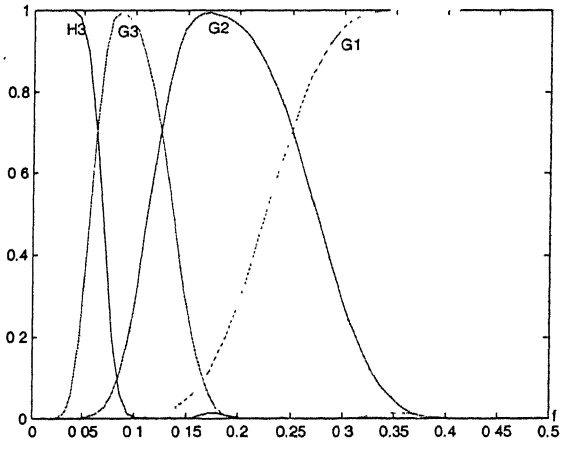
and therefore, provide a full coverage of the frequency domain.



(a)



(b)



(c)

Fig. 1. Frequency responses of the discrete filter bank associated with (a) 'db2' (b) 'db4'(c) 'db10'

2.2. Orthogonal Wavelet Bases of l_2

The sequence of discrete filters defined in the previous section 2.1 can be used to construct an orthogonal wavelet decomposition of l_2 (Rioul). The discrete normalized basis functions can be defined as

$$\varphi_{i,l}(k) = 2^{i/2} h_i(k-2^i l) \tag{2.6}$$

$$\psi_{i,l}(k) = 2^{i/2} g_i(k-2^i l) \tag{2.7}$$

Where i and l are the scale and translation indices, respectively; the factor $2^{i/2}$ is an inner product normalization. Now, we consider the sequence of nested subspaces $l_2 = V_0 \supset V_1 \supset \ldots \supset V_I$ where $V_i = \text{span}\{\varphi_{i,l}\}_{l \in Z}$ is the approximation space at resolution i. The embedding of these subspaces comes as a direct consequence of the two scale relations (2.4). We also introduce the subspaces W_i, (i = 1, 2, . . . , I), where W_i is the residue (or detail) space at resolution i and is defined as the orthogonal complement of V_i with respect to V_{i-1} (i.e., $V_{i-1} = V_i + W_i$ and $V_i \perp W_i$). It can be shown by induction that the families of sequences $\{\varphi_{i,l}\}_{l \in Z}$ and $\{\psi_{i,l}\}_{l \in Z}$ provide orthonormal bases of V_i and W_i, respectively. We can therefore evaluate the coordinates of the orthogonal projection of a discrete signal $x \in l_2$ in the spaces V_i and W_i by forming the inner product with corresponding basis function. The minimum l_2 – norm approximation of x at scale i (orthogonal projection into V_i) is given by

$$x_{(i)}(k) = \sum_{l \in z} s_{(i)}(l) \varphi_{i,l} \tag{2.8}$$

$$s_{(i)}(l) = \langle x(k), \psi_{i,l}(k) \rangle_{l_2} \tag{2.9}$$

where $<\cdot,\cdot>$ denotes the standard l_2 inner product and where $\varphi_{i,0}(k) = 2^{i/2} h_i(k)$ is the discrete scaling function at resolution i. The corresponding residue, which is the projection of x into W_i, is given by the complementary wavelet expansion

$$x_{(i-1)}(k) - x_{(i)}(k) = \sum_{l \in z} d_{(i)}(l)\psi_{i,l} \tag{2.10}$$

$$d_{(i)}(l) = <x(k), \psi_{i,l}(k)>_{l_2} \tag{2.11}$$

where $\psi_{i,0}(k) = 2^{i/2} g_i(k)$ is the discrete wavelet at scale i. Finally by combining the residues over all scales down to a given depth I, we obtain the full discrete wavelet expansion of a signal $x \in l_2$

$$x(k) = \sum_{l \in Z} s_{(I)}(l)\varphi_{I,l} + \sum_{i=1}^{I}\sum_{l \in Z} d_{(i)}(l)\psi_{i,l} \tag{2.12}$$

where $d_{(i)}$'s are the wavelet coefficients and $s_{(I)}$"s are the expansion coefficients of the coarser signal approximation $x_{(I)}$. It also follows from this construction that the family of sequences $\{\varphi_{i,l}, \psi_{1,l}, \psi_{2,l}, ..., \psi_{I,l}\}_{l \in Z}$ constitutes an orthonormal basis of l_2.

The analysis formulas 2.9 and 2.11 can be interpreted in terms of simple filtering and down-sampling operations. Specifically, we can write that

$$s_{(I)}(l) = 2^{I/2} [h_I^T * x]_{\downarrow 2^I}(l) \tag{2.13}$$

$$d_{(i)}(l) = 2^{i/2} [g_i^T * x]_{\downarrow 2^i}(l), \quad (i = 1,2,...,I) \tag{2.14}$$

where symbol ^{T} denoted the reversal operation (i.e., $h^T(k) = h(-k)$ and where $[\cdot]_{\downarrow m}$ is the down-sampling by a factor of m.

2.3. Discrete Wavelet Frames (DWF)

A simple integer shift of the input signal will usually result in a non-trivial modification of the discrete wavelet transform. As far as feature extraction is concerned, this behavior is inadequate, for one usually thinks of "texture" as a translation-invariant (or stationary) property. A natural way to over come this limitation is to perform an analysis of the input signal in terms of the overcomplete family of templates

$$S = \{g_1(k-l),\ldots,g_I(k-l),h_I(k-l)\}_{l\in Z} \tag{2.15}$$

which is similar to computing the discrete wavelet transform for all possible integer shifts of the input signal. This approach leads to the simple decomposition algorithm

$$\begin{aligned} d_i^{DWF}(k) &= < g_i(k-l),x(k) >_{l_2} = g_i^T * x(k) \\ S_I^{DWF}(k) &= < h_I(k-l),x(k) >_{l_2} = h_I^T * x(k) \\ &(i = 1,2,\ldots,I) \end{aligned} \tag{2.16}$$

which is non-sampled version of Equation(2.13) and Equation(2.14). Unlike other wavelet-based approaches, this overcomplete wavelet decomposition (the discrete wavelet frame (DWF)), in which the output of the filter bank is not sub sampled, result in a texture description that is invariant with respect to translations of the input signal. This property yields a better estimation of texture statistics and a more detailed texture characterization at region boundaries. Because of the special structure of the analysis filter bank, this decomposition has a number of remarkable properties that are associated with the mathematical concept of a frame [2]. These include energy conservation and a particularly simple reconstruction algorithm.

The reconstruction formula

$$x(k) = \sum_{l \in Z} s_I(l) h_I(k-l) + \sum_{i=1}^{I} \sum_{l \in Z} d_i(l) g_i(k-l) \tag{2.17}$$

which can also be interpreted in terms of a reconstruction filter bank.

$$x(k) = h_I * s_I(k) + \sum_{i=1}^{I} g_i * d_i(k) \tag{2.18}$$

2.4. 2D – Angular Wavelet Frames (AWF)

2.4.1. Extension of 1D approach to 2D

To extend the discrete wavelet frame (DWF) method to 2D, commonly used way is to use the standard tensor product formulation. This gives the decomposition, which is rectangular in nature in frequency domain as shown in Fig. 2. But our aim is to select dominant angular regions in the frequency plane.

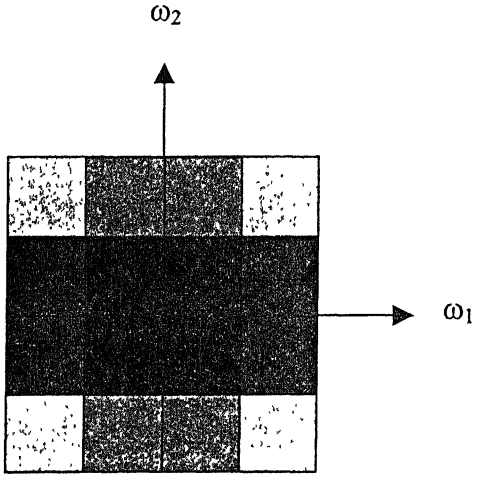


Fig. 2. Frequency plane segmentation in 2D separable wavelet frame

In order to overcome this problem we suggest frequency transformation method for the purpose of extension to 2D. In conventional 2D filter design techniques, frequency transformation methods are used to construct a 2D filter from 1D filter. We constructed 2D filters using two transformations – circular transformation and angular transformation on 1D filters. Different 1D prototype filters used in our work are 'db2', 'db4' and 'db10'. Now we describe the two transformation techniques here. Consider the transformation

$$H(\omega_1,\omega_2) = H(\omega)\,|_{\omega=T(\omega_1,\omega_2)} \tag{2.19}$$

where $T(\omega_1,\omega_2)$ is the transformation function. The frequency response of the 1D prototype filter $H(\omega)$ and the transformation function are jointly chosen to design 2D filter $H(\omega_1,\omega_2)$ with desired properties.

The two transformations that we have used are:

$$\begin{aligned} T(\omega_1,\omega_2) &= \sqrt{\omega_1^2+\omega_2^2} && - \textit{circular transformation} \\ &= 2*\text{atan}(\omega_2/\omega_1) && - \textit{angular transformation} \end{aligned} \tag{2.20}$$

The circular transformation on 1D low pass filter results in 2D circularly symmetric low pass filter. The low pass filters have significant response between $-\pi/2$ and $\pi/2$. On applying the 2D circular transformation on the low pass filter, a value of the low pass filter at a given ω is transferred onto a circle $\omega_1^2 + \omega_2^2 = \omega^2$ in the (ω_1, ω_2) plane. So the resulting 2D filter has significant response for the region $\omega_1^2 + \omega_2^2 < \pi/2$ and its magnitude beyond this region is very small. The frequency support of this 2D low pass filter is shown in Fig. 3. The region of support of this 2D filter is the shaded region in Fig. 3. The circular transformation on the corresponding

high pass filter results a 2D filter that has significant response for the region $\omega_1^2 + \omega_2^2 > \pi/2$ and its magnitude in the region $\omega_1^2 + \omega_2^2 < \pi/2$ is very small. The region of support of this 2D filter is the unshaded region in Fig. 3.

The angular transformation on 1D low pass filter gives a 2D angular filter. On applying the 2D angular transformation on the low pass filter, a value of the low pass filter at a given ω is transferred onto a straight line $\omega_2 = \omega_1 \tan(\omega/2)$ in (ω_1, ω_2) plane. For $0 < \omega < \pi/2$, and for positive values of ω_1, H (ω) values are transferred to the angular region $0 < \theta < \pi/4$ and for negative values of ω_1 they are transferred to the region $\pi < \theta < 5\pi/4$ of the frequency plane. For $-\pi/2 < \omega < 0$, and for positive values of ω_1, H (ω) values are transferred to the angular region $-\pi/4 < \theta < 0$ and for negative values of ω_1 they are transferred to the region $3\pi/4 < \theta < \pi$ of the frequency plane. This 2D filter is hermitian symmetric. The frequency support of angularly transformed low pass filter is the shaded region in Fig. 4. The angular transformation of the corresponding high pass filter results in a 2D filter, which has frequency support in the unshaded region in Fig. 4.

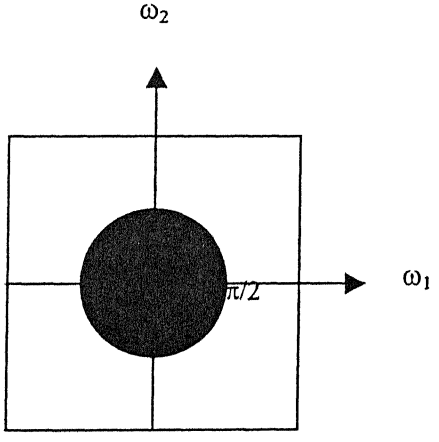


Fig. 3. Circular Transformation

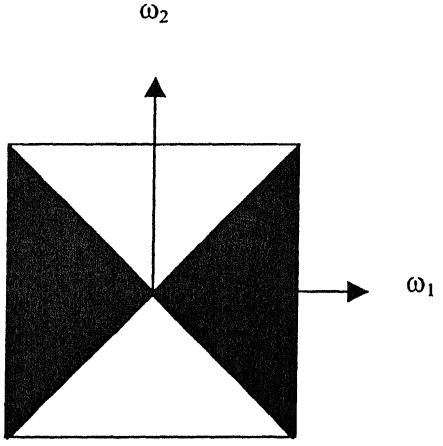


Fig. 4. Angular Transformation

2.4.2. Construction of 2D Angular Wavelet Frames

2.4.2.1. Decomposition at Level 1

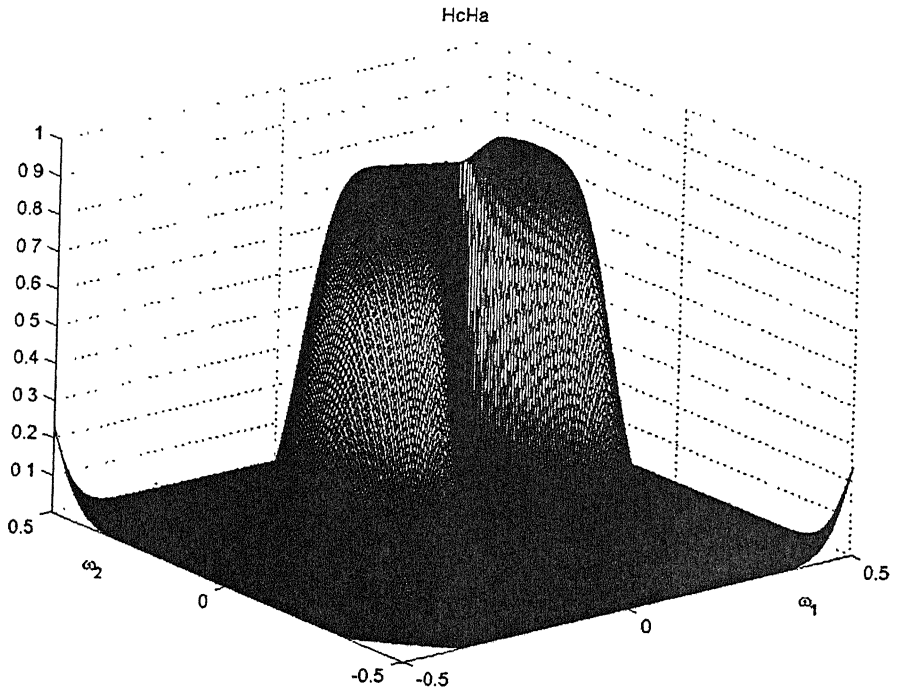
Consider 1D PRFB filters $H(\omega)$ and $G(\omega)$ to construct a 2D angular wavelet frame based on 1D wavelet frame. On applying the circular and angular transformations described in Equation (2.20) we obtain four filters H_c, G_c, H_a and G_a.

$$\begin{aligned} H_c(\omega_1,\omega_2) &= H(\omega)\big|_{\omega=\sqrt{\omega_1^2+\omega_2^2}} \\ G_c(\omega_1,\omega_2) &= G(\omega)\big|_{\omega=\sqrt{\omega_1^2+\omega_2^2}} \\ H_a(\omega_1,\omega_2) &= H(\omega)\big|_{\omega=2*a\tan(\omega_2/\omega_1)} \\ G_a(\omega_1,\omega_2) &= G(\omega)\big|_{\omega=2*a\tan(\omega_2/\omega_1)} \end{aligned} \tag{2.21}$$

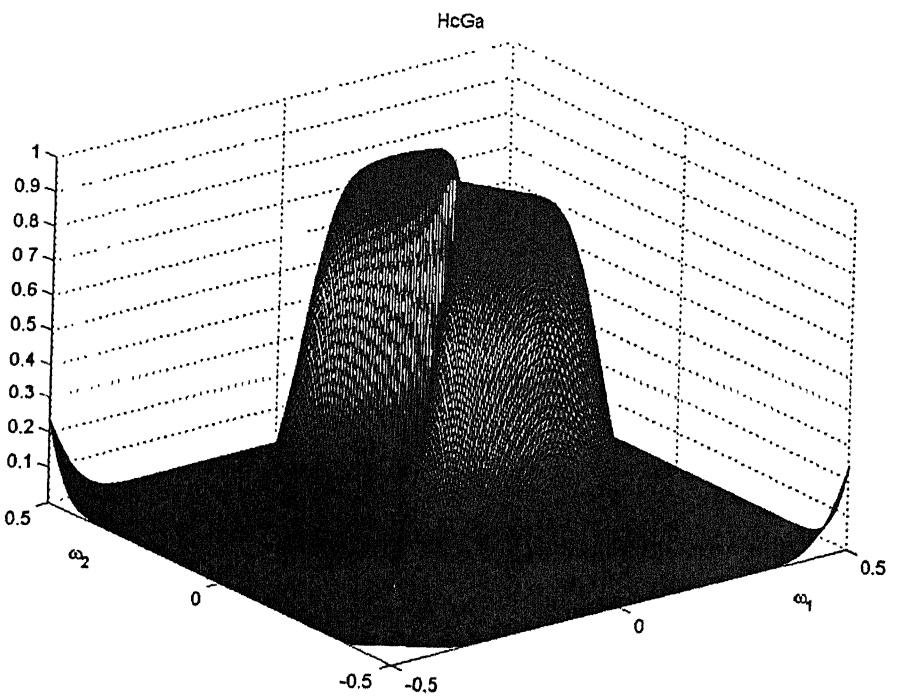
where the subscript 'c' and 'a' denotes circular and angular transformations respectively. A useful wavelet frame can be constructed with the analysis filters

$$F_{hh} = H_cH_a;\ F_{hg} = H_cG_a;\ F_{gh} = G_cH_a;\ F_{gg} = G_cG_a; \tag{2.22}$$

The responses of these four filters using 1D prototype filter 'db10' are shown in Fig. 5. The Region of Supports (ROS) of these filters are shown in the Fig. 6. Fig. 6 (a) shows the ROS of HcHa, Fig. 6 (b) shows the ROS of HcGa, Fig. 6 (c) shows the ROS of GcHa, Fig. 6 (d) shows the ROS of GcGa. It is evident that the constructed 2D angular wavelet frames can capture the information in the regions as shown in the Fig. 5 and Fig. 6. At the synthesis end, the conjugate filters are constructed using conjugates of the 1D filters in similar steps. The principle of this decomposition is illustrated in Fig. 7. This whole system of filters acts as the identity operator, i.e., the analysis and synthesis filters constitute a perfect reconstruction filter bank

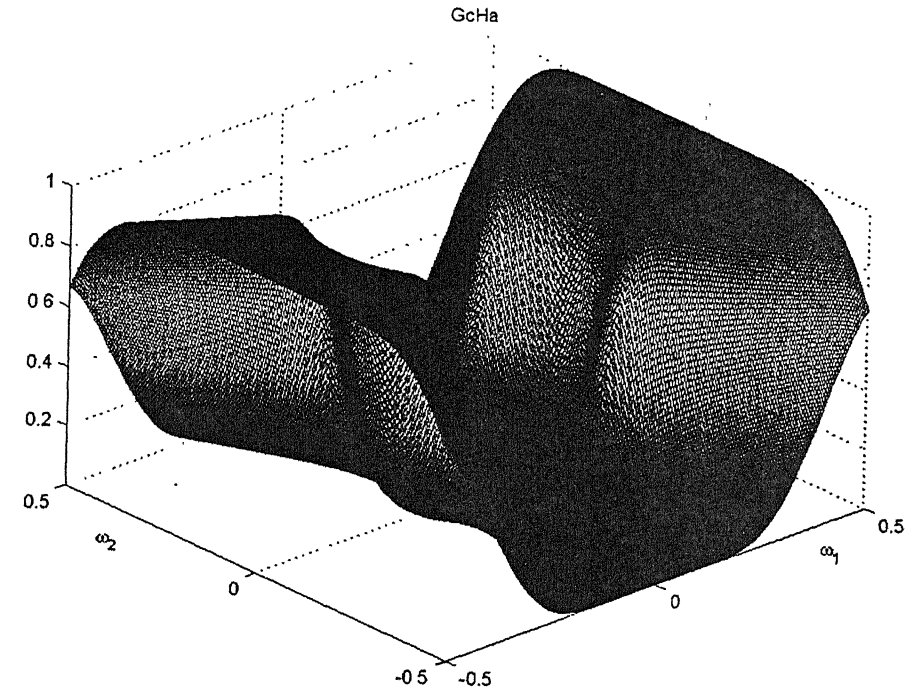


(a)

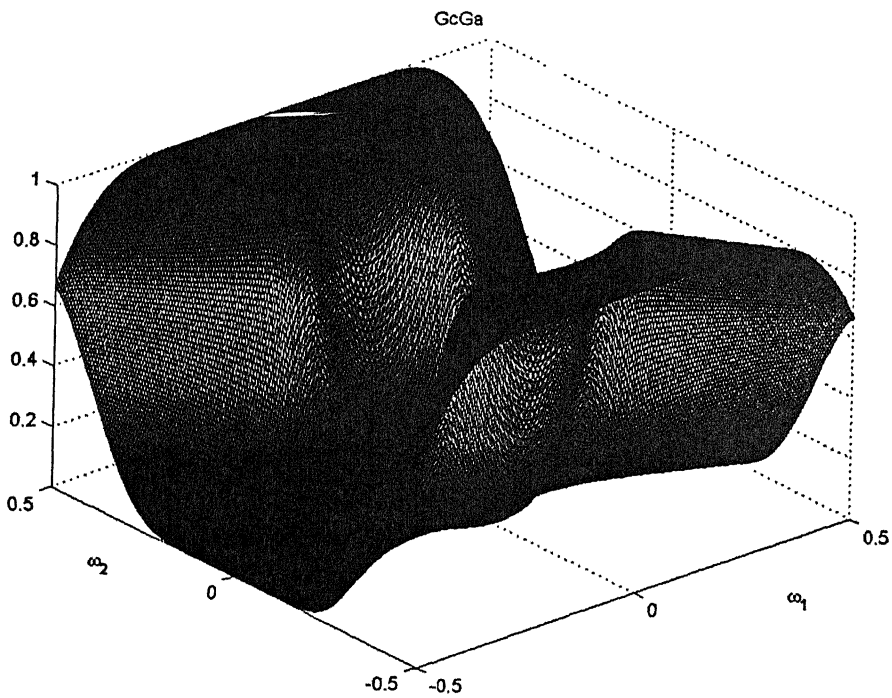


(b)

Fig. 5. Filter responses for level 1 using 'db10' (a) HcHa (b) HcGa



(c)



(d)

Fig. 5. Filter responses for level 1 using 'db10' (c) GcHa (d) GcGa

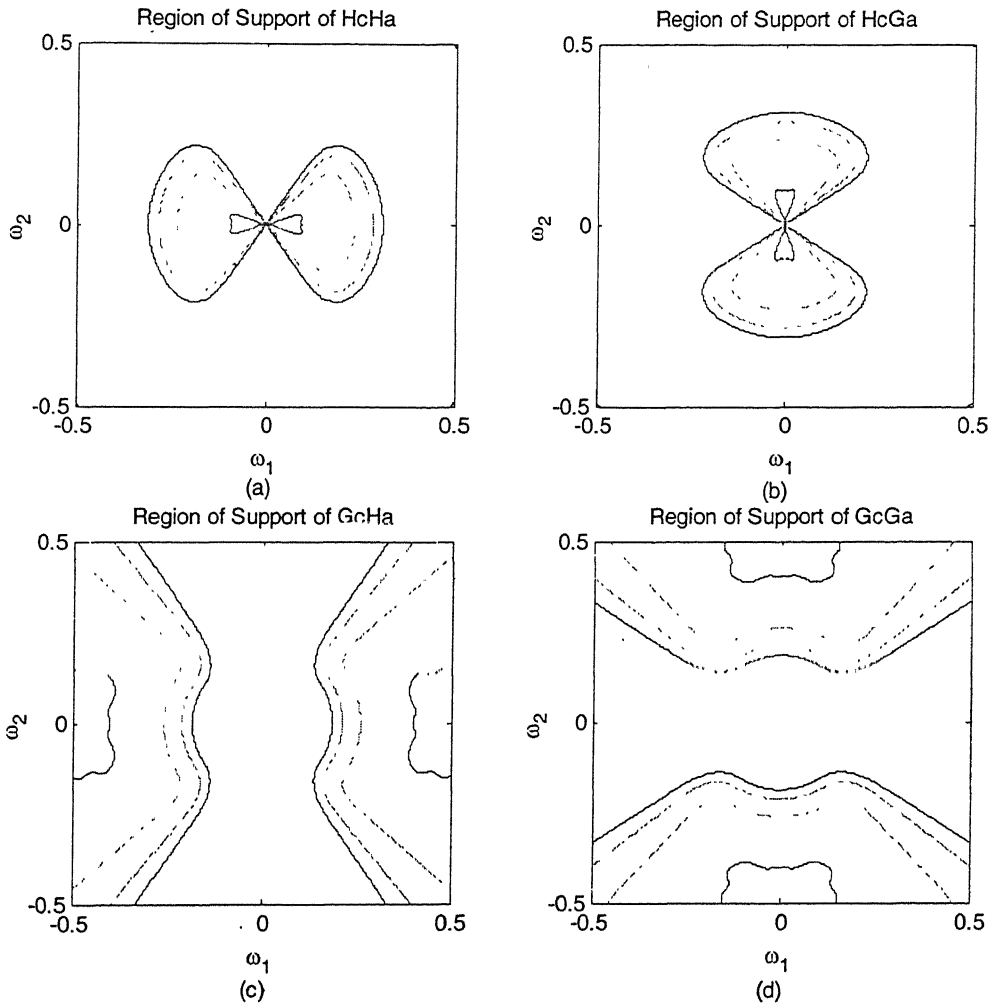


Fig. 6. Region of Support of the one level Angular Wavelet Frames using 'db10'

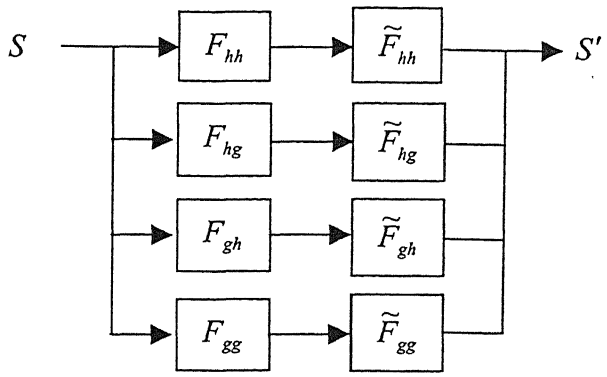


Fig. 7. Single level Signal analysis and synthesis using the Angular Wavelet Frame decomposition

Property: The 2D analysis filters constructed by circular and angular transformations of 1D PRFB along with their conjugates at the synthesis end constitute a perfect reconstruction filter bank.

Proof: From the Fig. 7, the output signal in terms of input can be written as

$$S' = \left[F_{hh}\,\tilde{F}_{hh} + F_{hg}\,\tilde{F}_{hg} + F_{gh}\,\tilde{F}_{gh} + F_{gg}\,\tilde{F}_{gg}\right] S$$

using eqn.(2.22)

$$S' = \left[H_c\,H_a\tilde{H}_c\tilde{H}_a + H_c\,G_a\tilde{H}_c\tilde{G}_a + G_c\,H_a\tilde{G}_c\tilde{H}_a + G_c\,G_a\tilde{G}_c\tilde{G}_a\right]S$$

$$= \left[H_c\,\tilde{H}_c + G_c\,\tilde{G}_c\right]\left[H_a\,\tilde{H}_a + G_a\,\tilde{G}_a\right]S$$

$$= S \text{ (using PRFB condition eqn.(2.1)}$$

Hence, the condition of perfect reconstruction for the designed filter bank is satisfied.

2.4.2.2. Decomposition at Higher Level

Now we describe the method to decompose the signal at higher level. First consider the pyramidal decomposition where the low-low region is again decomposed into four regions. The set of 1D prototype filters for the higher level of decomposition are obtained using the Equation (2.3). For level 2, using H_2 and G_2 (i=1) 2D circular and angular transformed filters H_{2c} , G_{2c} , H_{2a} and G_{2a} are obtained.

Consider the following combination of analysis filters:

$$F_{2hh} = H_{2c}\, H_{2a};\ F_{2hg} = H_{2c}\, G_{2a};\ F_{2gh} = G_{2c}\, H_{2a};\ F_{2gg} = G_{2c}\, G_{2a}; \qquad (2.23)$$

These four filters decompose the region supported by F_{hh} (HcHa) into four regions. The frequency responses of these four filters are shown in Fig. 8. The Region of Support of these filters is shown in Fig. 12. These four filters along with H_cG_a, G_cH_a and G_cG_a are used for the second level of decomposition. At the synthesis end the conjugate filters are constructed using the conjugates of 1D filters in similar steps. These seven filters along with their conjugates at the synthesis end constitute a perfect reconstruction filter bank.

The output in terms of input can be written as:

$$S' = \left[F_{2hh}\,\widetilde{F}_{2hh} + F_{2hg}\,\widetilde{F}_{2hg} + F_{2gh}\,\widetilde{F}_{2gh} + F_{2gg}\,\widetilde{F}_{2gg} + F_{hg}\,\widetilde{F}_{hg} + F_{gh}\,\widetilde{F}_{gh} + F_{gg}\,\widetilde{F}_{gg}\right]S$$

Using Equations (2.23), (2.3) and (2.1) we get

$$S' = \left[H_c\widetilde{H}_c H_a\widetilde{H}_a + F_{hg}\,\widetilde{F}_{hg} + F_{gh}\,\widetilde{F}_{gh} + F_{gg}\,\widetilde{F}_{gg}\right]S$$

Using Equations (2.22) and (2.23) we get

$$S' = S$$

Hence the condition of perfect reconstruction for the designed filter bank is satisfied.

Using the pyramidal decomposition we obtain N = 1 + 3I (where I is the level of decomposition.) number of filters.

We can as well decompose the other regions of the frequency plane each into four regions, which is called tree structured decomposition method. Using Equation (2.3) we can get decomposition in the low-low frequency region only. In order to decompose the other regions, we modified the Equation (2.3). We describe this method for the case of level 2. Consider the following four equations:

$$H1H2(z) = H(z^2)H(z) \tag{2.24}$$

$$H1G2(z) = G(z^2)H(z) \tag{2.25}$$

$$G1H2(z) = H(z^2)G(z) \tag{2.26}$$

$$G1G2(z) = G(z^2)G(z) \tag{2.27}$$

The first two equations are same as Equation (2.3) for i=1. Using circular and angular transformations on these four 1D filters we obtain 2D filters $H1H2_c$, $H1G2_c$, $G1H2_c$, $G1G2_c$, $H1H2_a$, $H1G2_a$, $G1H2_a$ and $G1G2_a$.

To decompose in the low-low region the set of filters to be used are:

$$F_1 = H1H2_c\ H1H2_a;\ F_2 = H1H2_c\ H1G2_a;\ F_3 = H1G2_c\ H1H2_a;\ F_4 = H1G2_c\ H1G2_a; \tag{2.28}$$

These four filters are same as the filters given by Equation (2.23). $F_1 = F_{2hh}$; $F_2 = F_{2hg}$; $F3 = F_{2gh}$; $F_4 = F_{2gg}$;

To decompose in the low-high region the set of filters to be used are:

$$F_5 = H1H2_c\ G1G2_a;\ F_6 = H1H2_c\ G1H2_a;\ F_7 = H1G2_c\ G1G2_a;\ F_8 = H1G2_c\ G1H2_a; \quad (2.29)$$

The frequency responses of these four filters are shown in Fig. 9. The Region of Support of these filters is shown in Fig. 13.

To decompose in the high-low region the set of filters to be used are:

$$F_9 = G1G2_c\ H1H2_a;\ F_{10} = G1G2_c\ H1G2_a;\ F_{11} = G1H2_c\ H1H2_a;\ F_{12} = G1H2_c\ H1G2_a; \quad (2.30)$$

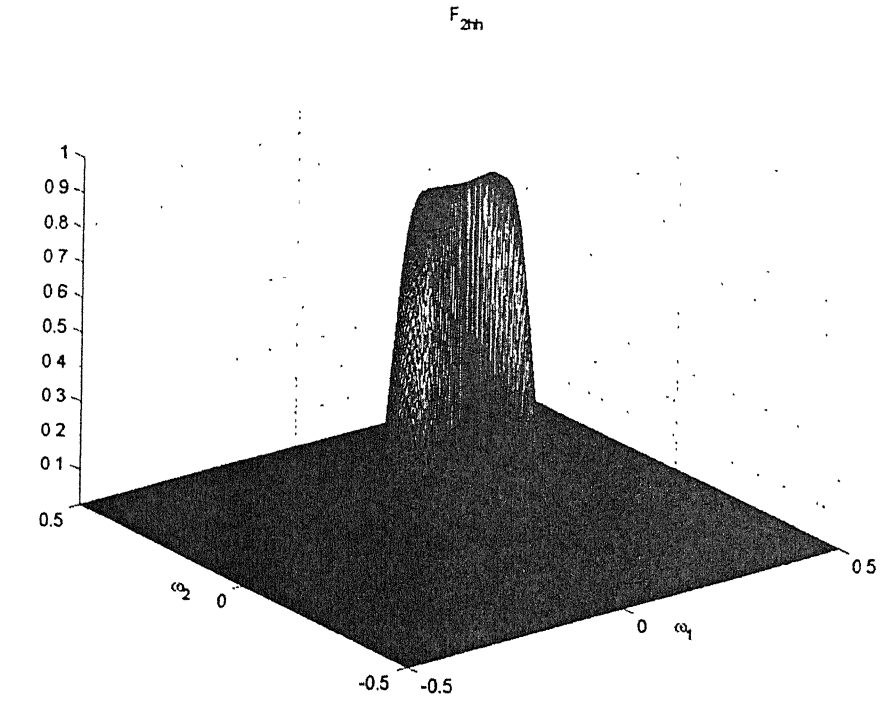
The frequency responses of these four filters are shown in Fig. 10. The Region of Support of these filters is shown in Fig. 14.

To decompose in the high-high region the set of filters to be used are:

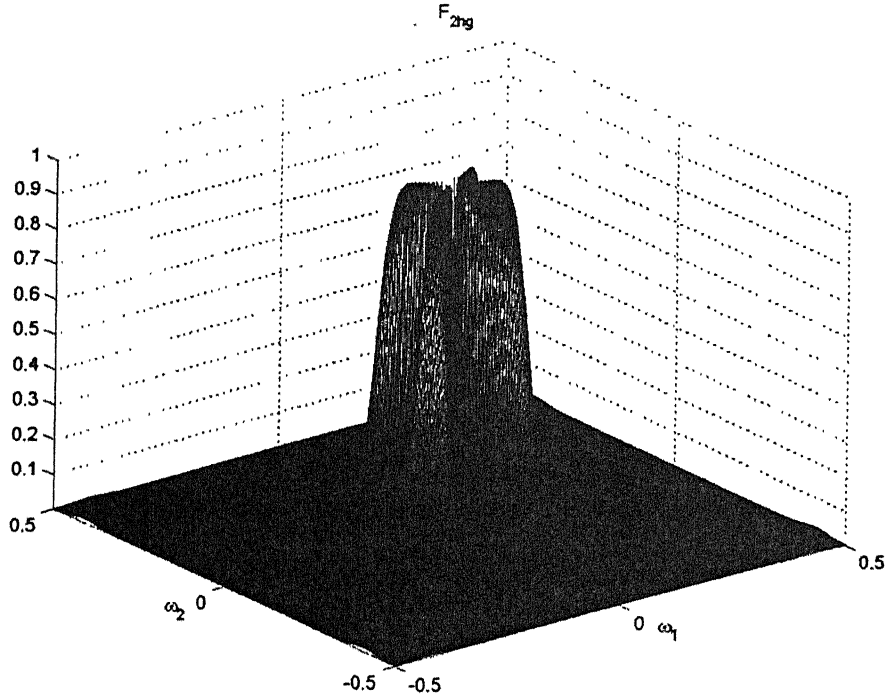
$$F_{13} = G1G2_c\ G1G2_a;\ F_{14} = G1G2_c\ G1H2_a;\ F_3 = G1H2_c\ G1G2_a;\ F_2 = G1H2_c\ G1H2_a; \quad (2.31)$$

The frequency responses of these four filters are shown in Fig. 11. The Region of Support of these filters is shown in Fig. 15.

At the synthesis end the conjugate filters are constructed using the conjugates of 1D filters in similar steps. The filter bank designed for the analysis along with their conjugates at the synthesis end constitutes a perfect reconstruction filter bank. In the next chapter, the application of these 2D Angular Wavelet Frames for texture classification and segmentation is presented.

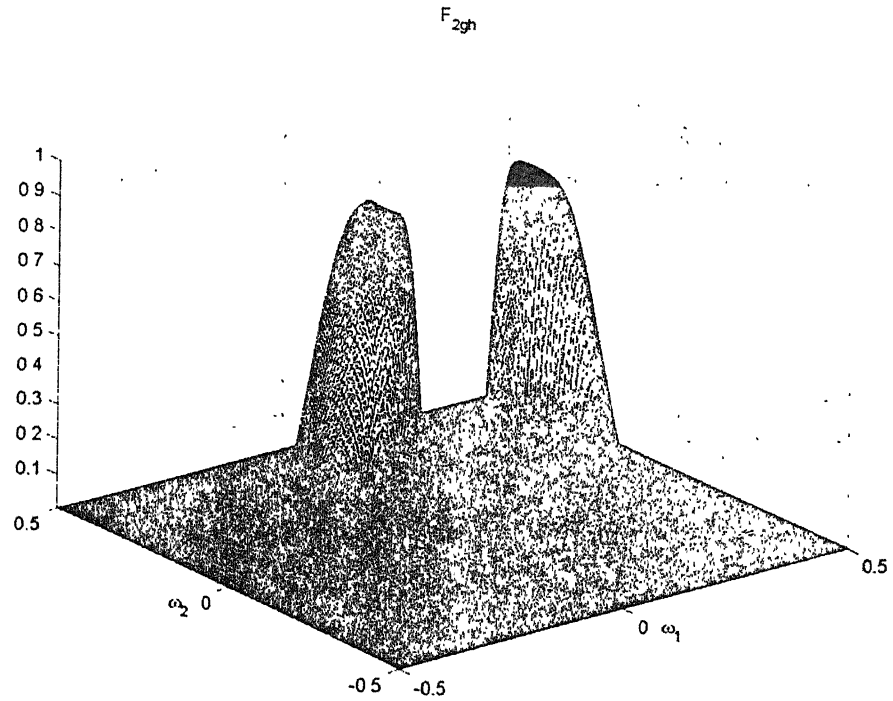


(a)

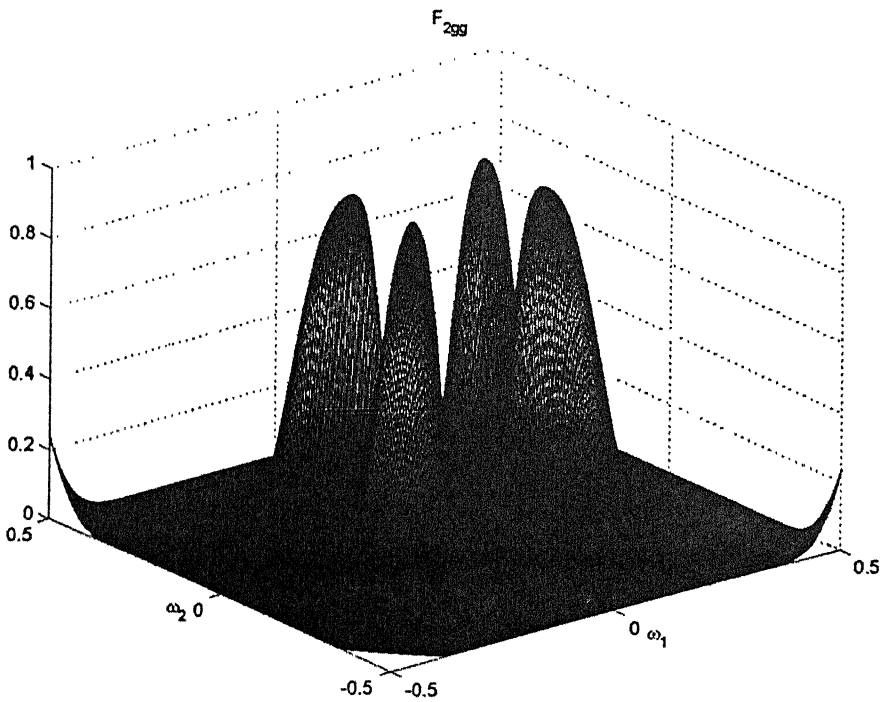


(b)

Fig. 8. Frequency responses of filters (a) F_{2hh} (b) F_{2hg}

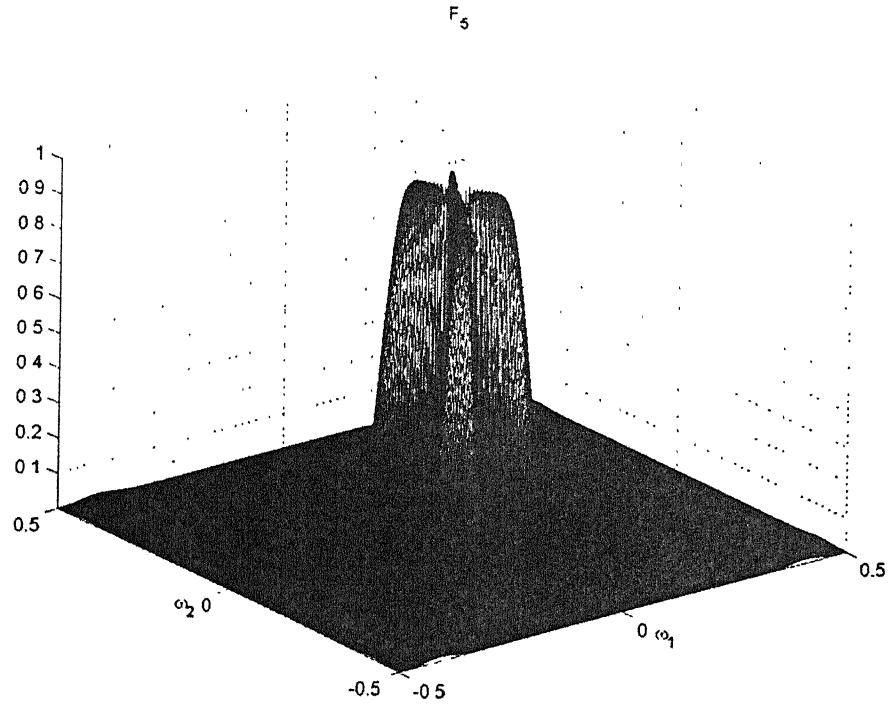


(c)

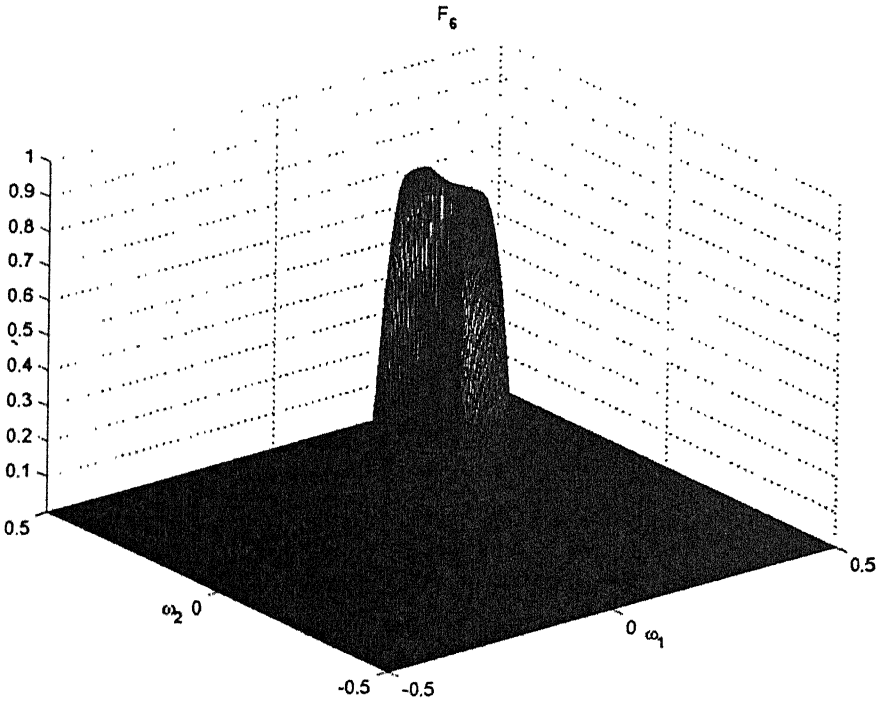


(d)

Fig. 8. Frequency responses of filters (a) F_{2gh} (b) F_{2gg}

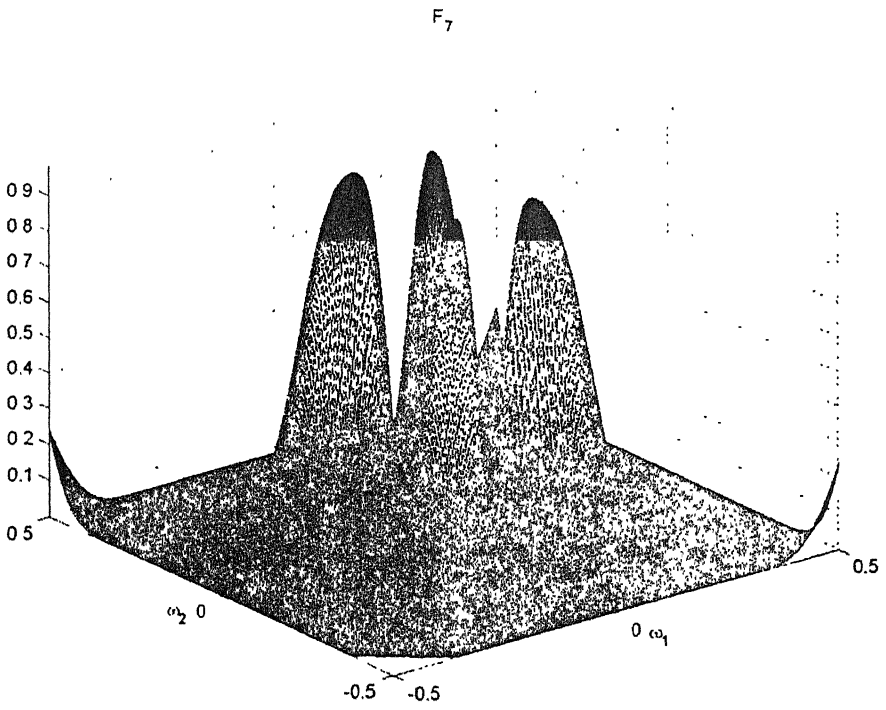


(a)

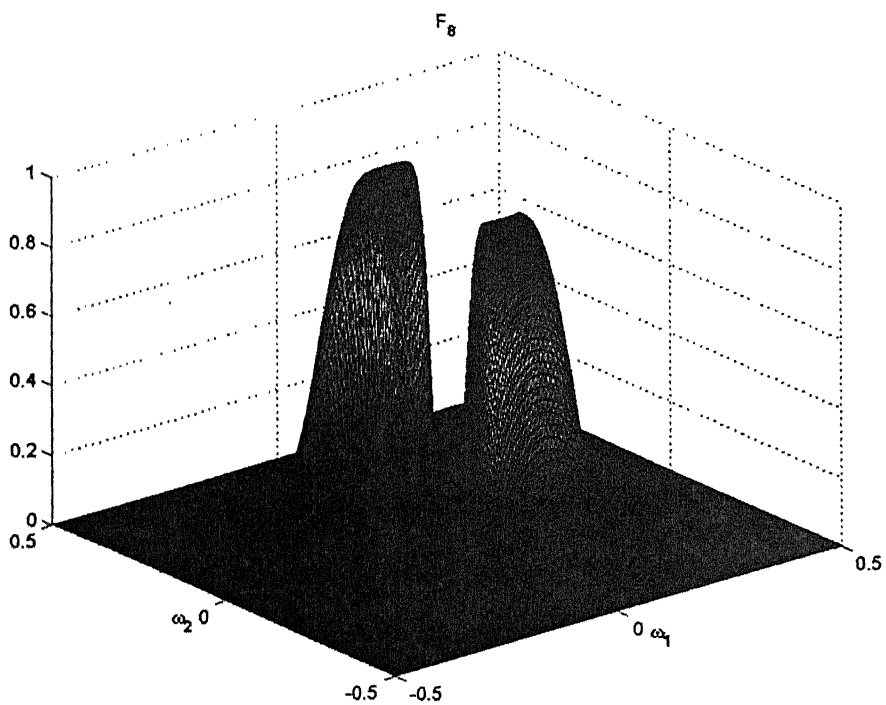


(b)

Fig. 9. Frequency responses of filters (a) F_5 (b) F_6

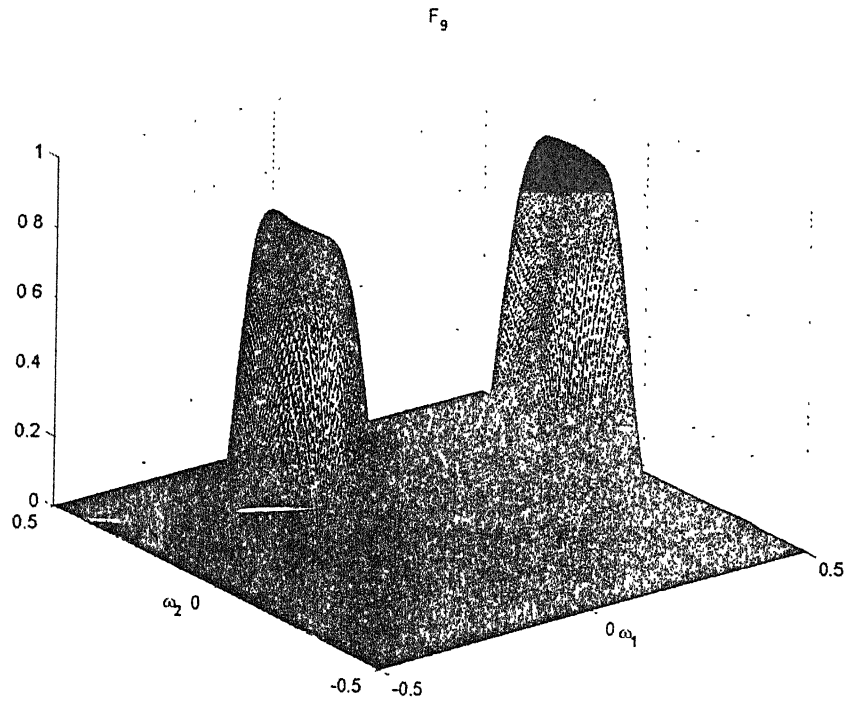


(c)

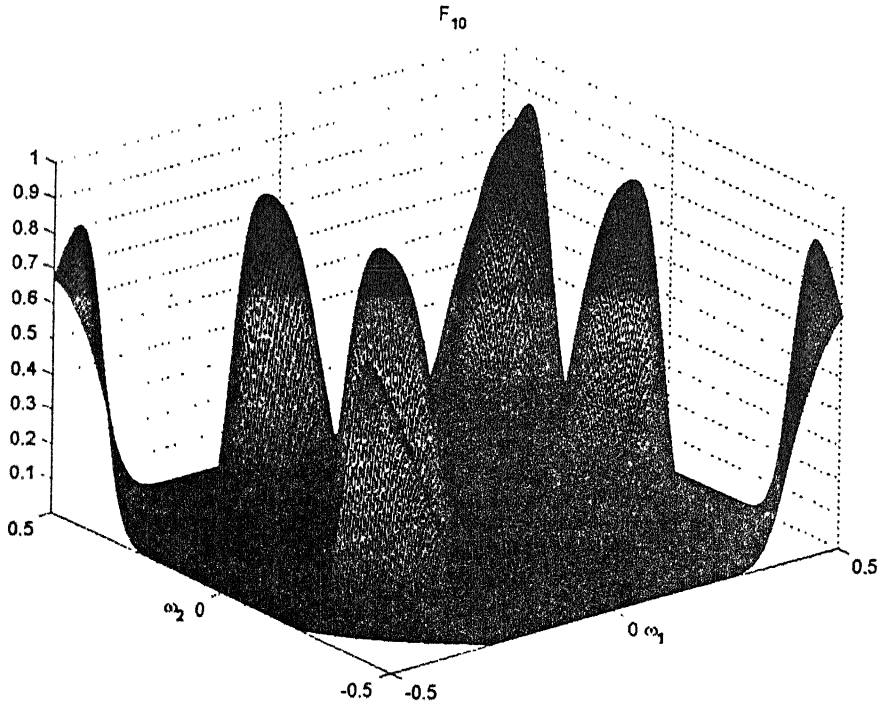


(d)

Fig. 9. Frequency responses of filters (c) F_7 (d) F_8

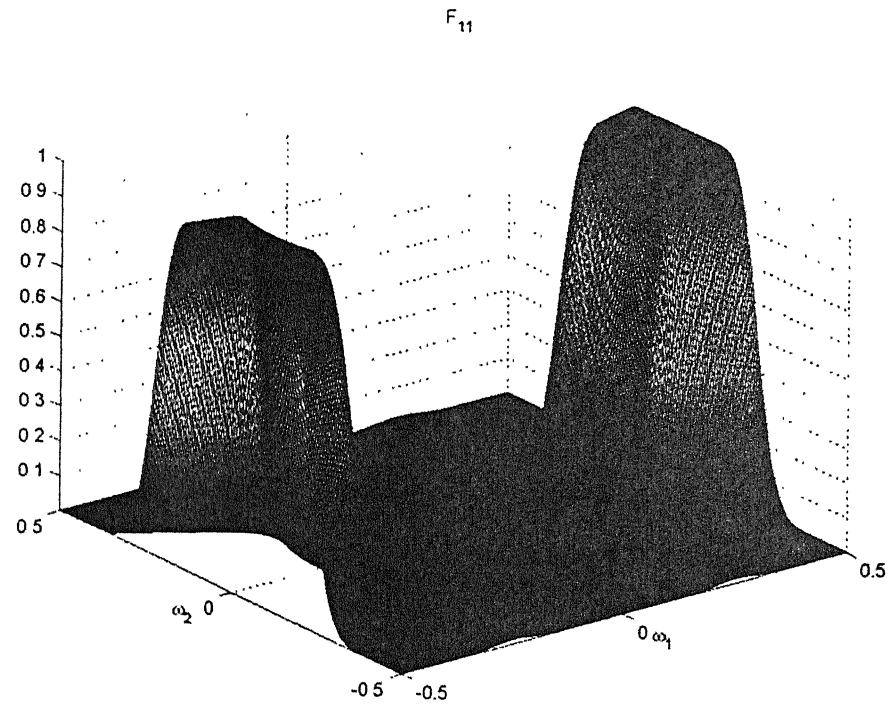


(a)

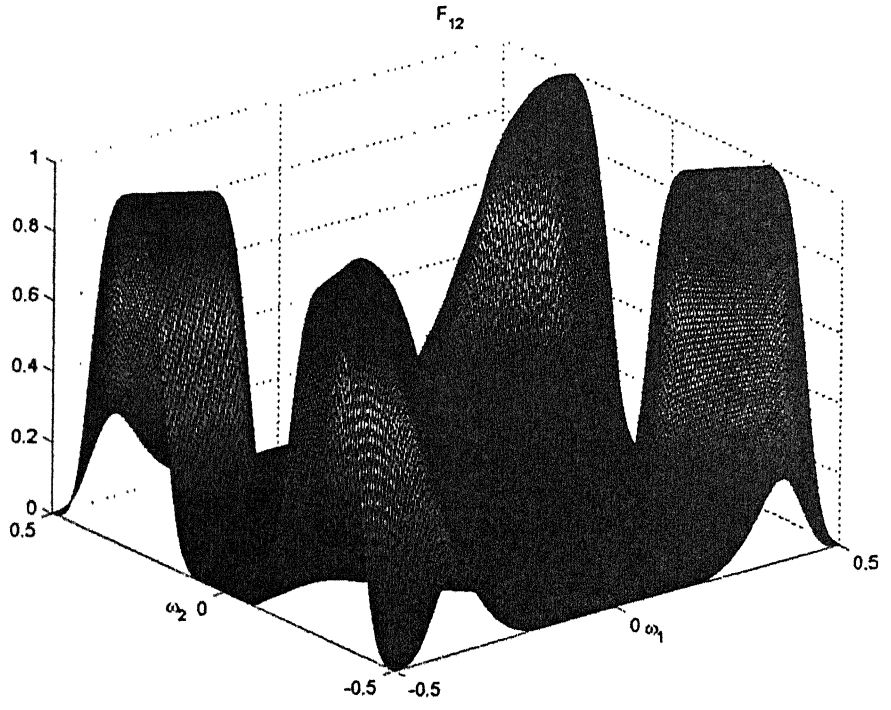


(b)

Fig. 10. Frequency responses of filters (a) F_9 (b) F_{10}

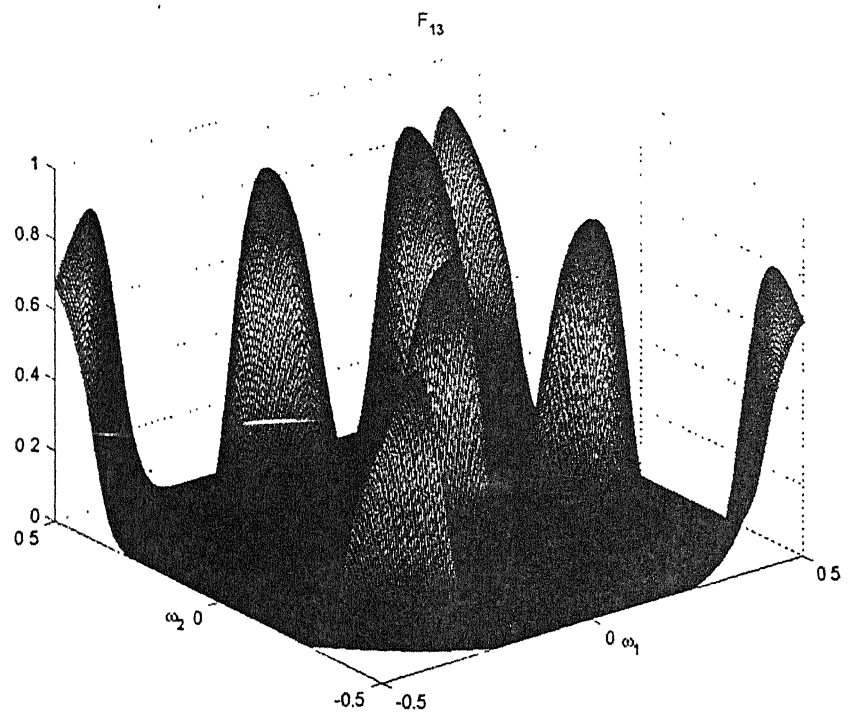


(c)

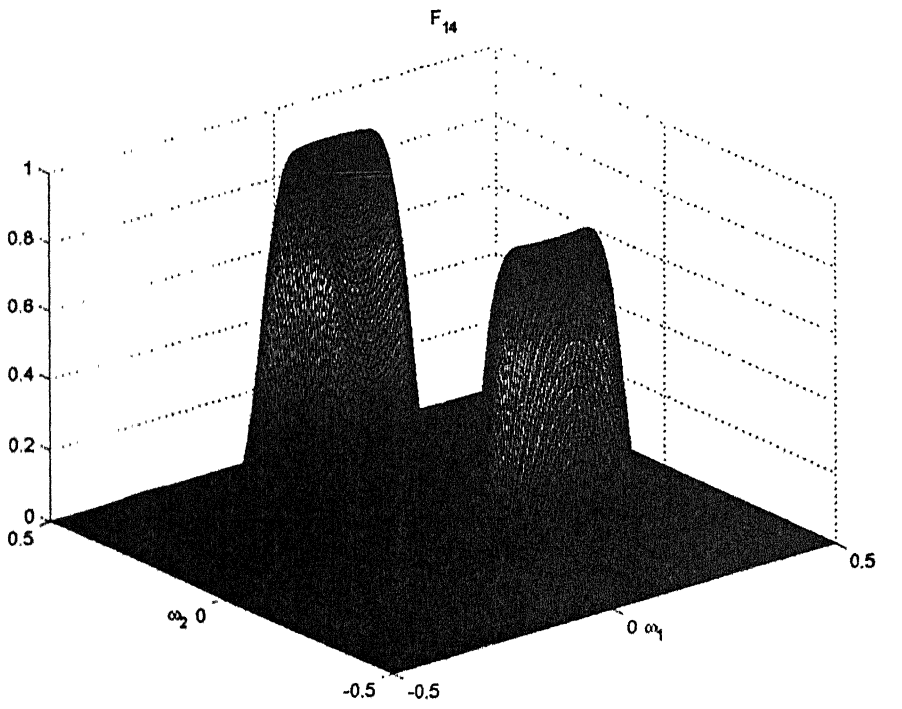


(d)

Fig. 10. Frequency responses of filters (c) F_{11} (d) F_{12}

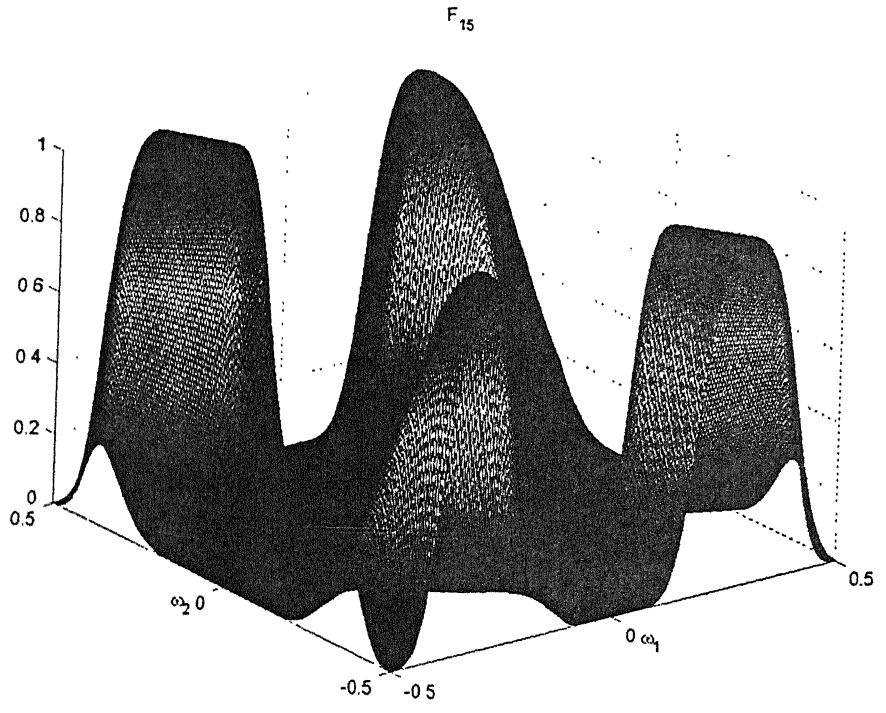


(a)

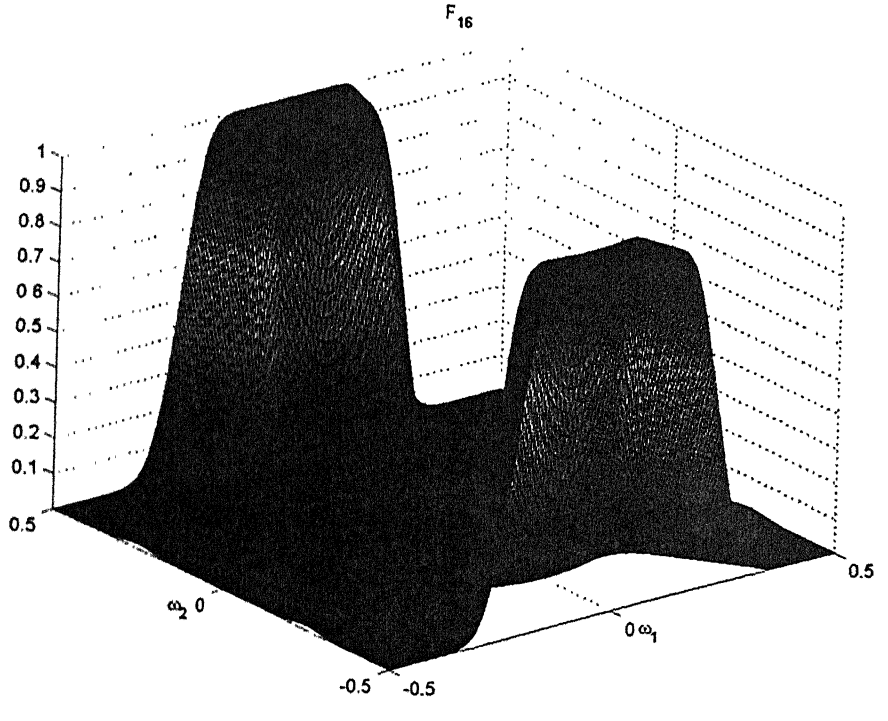


(b)

Fig. 11. Frequency responses of filters (a) F_{13} (b) F_{14}



(c)



(d)

Fig. 11. Frequency responses of filters (c) F_{15} (d) F_{16}

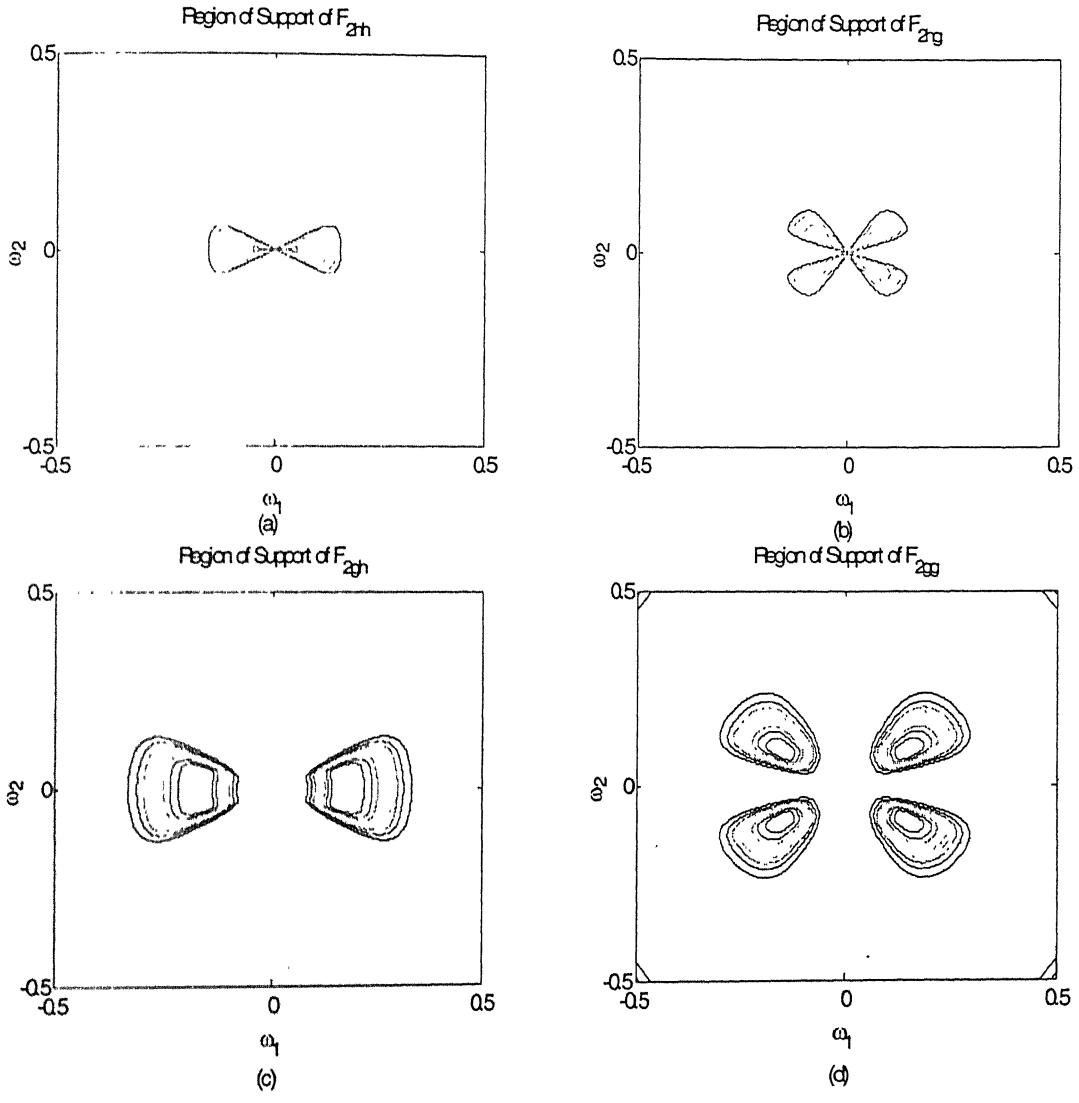


Fig. 12. Region of Support of (a) F_{2hh} (b) F_{2hg} (c) F_{2gh} (d) F_{2gg}

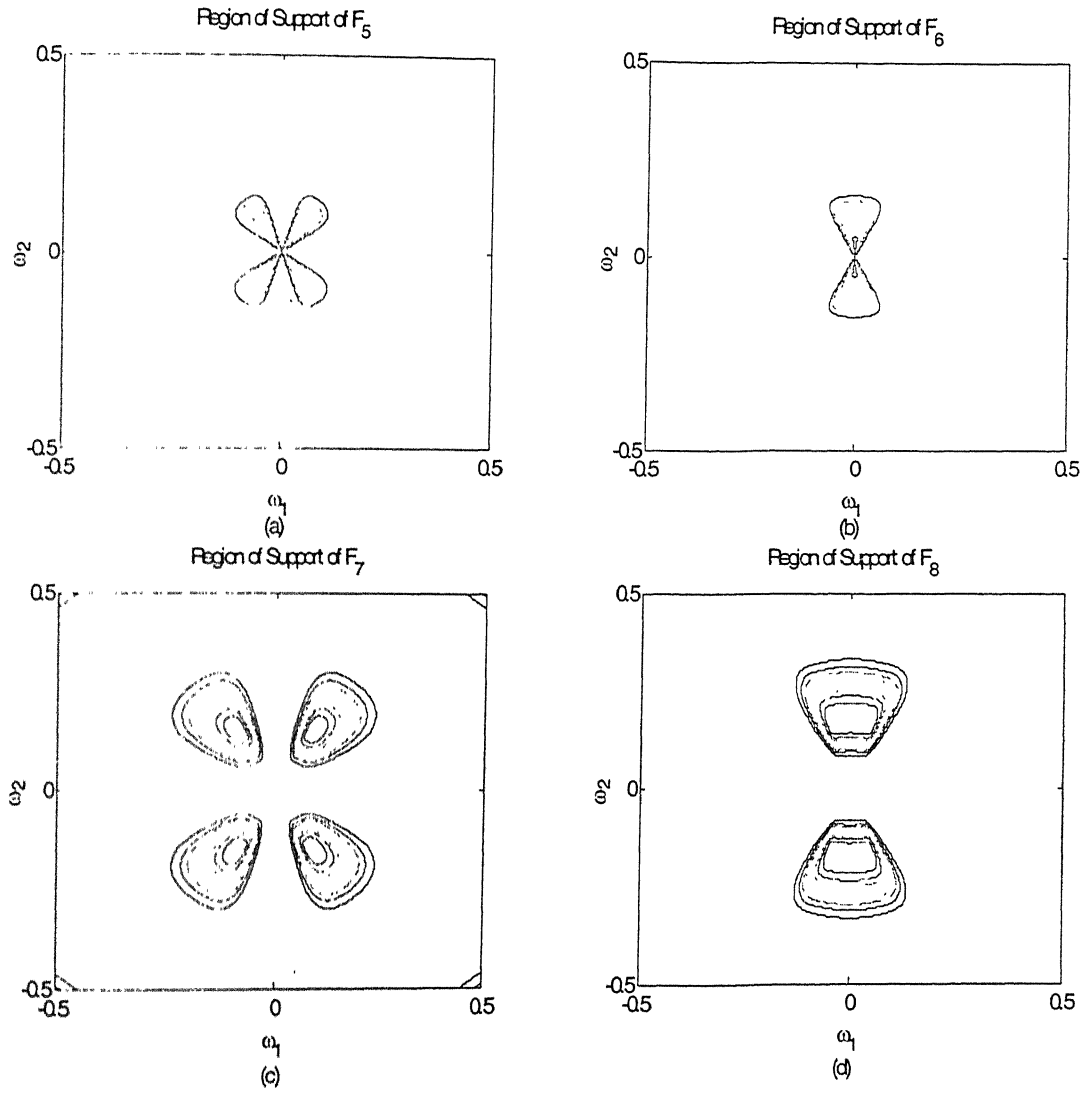


Fig. 13. Region of Support of (a) F_5 (b) F_6 (c) F_7 (d) F_8

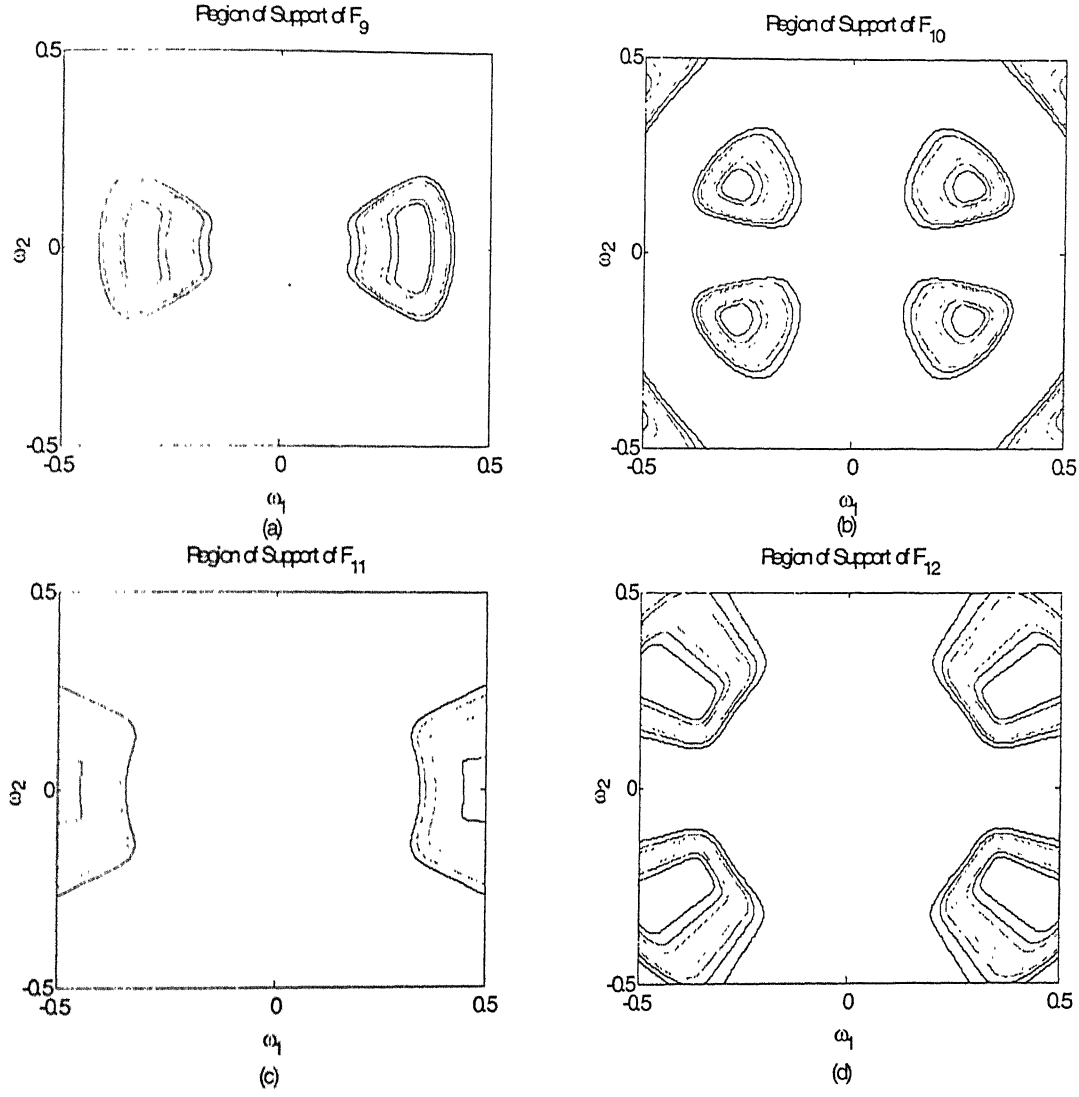


Fig. 14. Region of Support of (a) F_9 (b) F_{10} (c) F_{11} (d) F_{12}

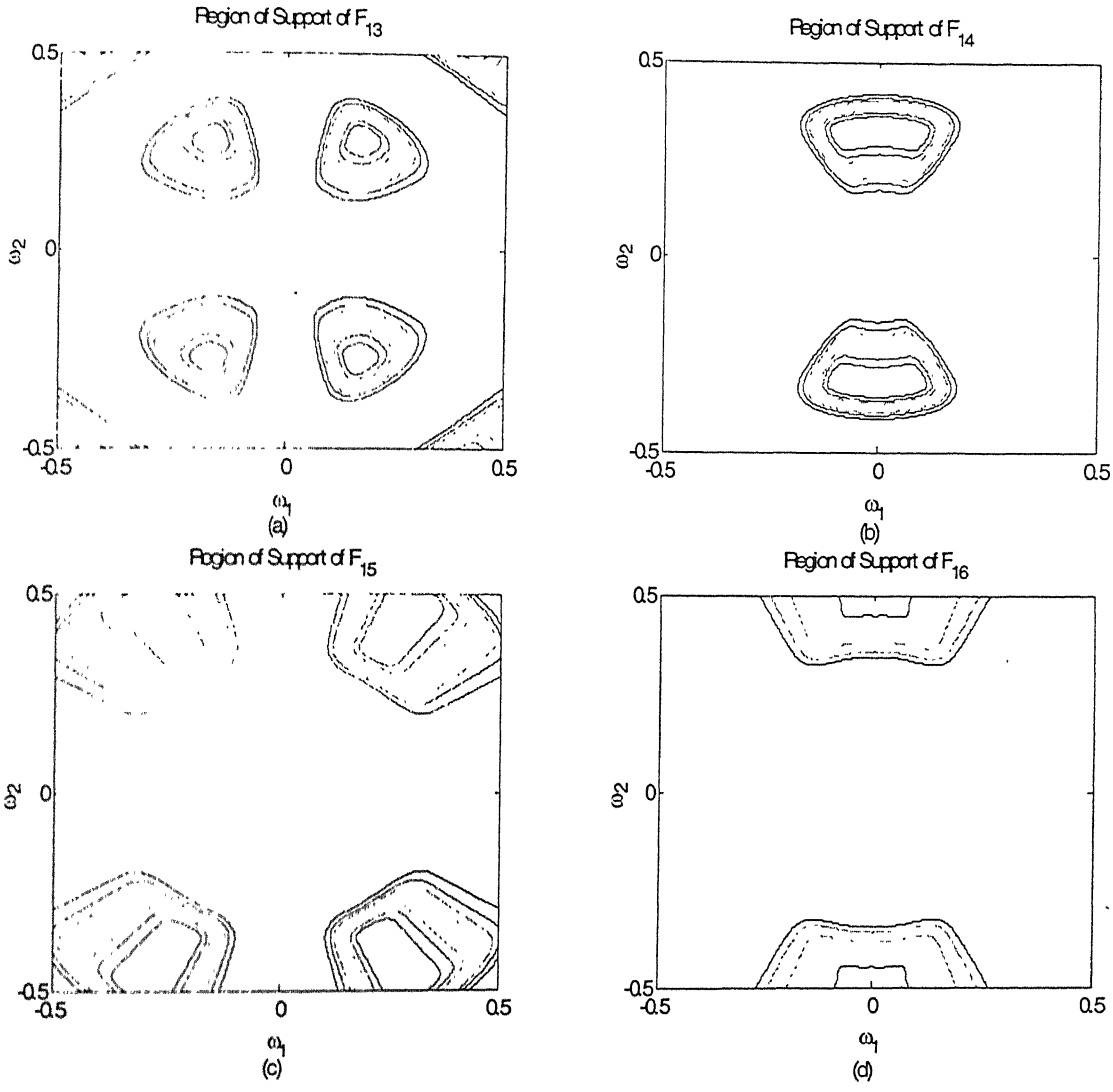


Fig. 15. Region of Support of (a) F_{13} (b) F_{14} (c) F_{15} (d) F_{16}

Chapter 3

TEXTURE CLASSIFICATION AND SEGMENTATION

3.1. TEXTURE CLASSIFICATION

Texture classification proceeds in two phases: characterization - where features are extracted from the textured input image, and the classification – where the feature vectors of a given textured input image are assigned to a class using a classification algorithm.

3.1.1. Texture Characterization

Here we arrange the output of the filter bank into N-component vector, where N is the number of subbands,

$$\begin{aligned} \mathbf{y}(k,l) &= (y_i(k,l))_{i=1,\ldots N} \\ &= [s_I(k,l) \;\; d_I(k,l) \;\; \ldots \;\; d_1(k,l)]^T \end{aligned} \tag{3.1}$$

And interpret the result of the analysis for a given spatial index (k, l) as a local linear transformation of the input vector x (k, l). A 2-D pyramidal angular wavelet transform with a depth I yields N = 1 + 3*I such feature channels. The texture is then characterized by the set of

N first-order probability density functions $p(y_i)$, i=1,...N. Alternatively; we can get a more compact representation in terms of the channel variances $Var\{y_i\}$. The success of this method obviously depends on the judicious selection of the filter bank. In practice, the channel variances are estimated from the average sum of squares over a region of interest R of the given texture type.

$$v_i = \frac{1}{\#R} \sum_{(k,l)\in R} y_i^2(k,l) \tag{3.2}$$

where #R denoted the number of pixels in R. We can get a better variance estimate by subtracting E $\{x\}^2$ in the lowpass channel.

3.1.2. Texture Classification

3.1.2.1. Experimental Data Set

We performed classification experiments using 26 Brodatz textures. The images taken are of size 256*256. For each texture, feature vectors $\mathbf{v} = (v_1, v_2, ...v_N)$ were evaluated, using eqn (3.2) using a window of size w*w over non-overlapping sub regions. The different window sizes that we have taken are 8*8, 16*16 and 32*32.

3.1.2.2. Classification Algorithm

The class conditional probability density functions were assumed to be multivariate Gaussian with mean vectors and covariance matrices ($\mathbf{m}_k$, $\mathbf{C}_k$), (k = 1, 2, ... ,K), where K = 26 is the total

number of classes. Under such an assumption, the minimum error Bayes Classifier is equivalent to assigning a texture sample with feature vector **v** to the class with minimum distance value

$$d_k(\mathbf{v}) = (\mathbf{v} - \mathbf{m}_k)^T \mathbf{C}_k^{-1}(\mathbf{v} - \mathbf{m}_k) + log(det(\mathbf{C}_k)) \quad (3.3)$$

For each pattern tested, the training was performed on the remaining set (leaving-one-out method) using the maximum likelihood estimates of the distribution parameters.

3.2. TEXTURE SEGMENTATION

Our next objective is to modify the previous wavelet based feature extraction method to make it suitable for texture segmentation. It proceeds in two steps: feature extraction – where features are extracted from the textured input image and the optimization stage – in this stage feature vectors are grouped into the homogenous segments by minimizing an appropriate quality measure.

3.2.1. Feature Extraction

Texture features provide a higher order description of the local image content. The segmentation algorithm, which is capable of accurately recognizing the patterns, relies on good features. By good features we mean that the patterns of the same class yield similar feature vectors. Features are extracted from filtered images. The texture features that are used for segmentation are local energy measures computed over a window centered on the current spatial location. The feature image $E_k(x, y)$ corresponding to the filtered image $f_k(x, y)$ is given by:

$$E_k(x,y) = \frac{1}{N^2} \sum_{(a,b)\in W_{xy}} |f_k(a,b)|^2 \qquad (3.4)$$

where W_{xy} is an N*N window centered at the pixel with coordinates (x, y). The result of averaging is a "feature image" in which the gray level at each pixel is a measure of the texture present at the corresponding location of the original image. When computing the texture features for pixels near the image boundary we assume that the image is extended by its mirror image. The size, N, of the averaging window in eqn (3.4) is an important parameter. As the window size increases, the computed value of the features at a point is determined by the large neighborhood of the image pixels, so that it gives more reliable values. But in real texture segmentation problems, the region size and boundaries are unknown. The large window size can significantly overlap regions and blurs the boundaries between textured regions. So for accurate determination of texture boundary, the smaller window size is needed. Therefore, there is a trade-off between reliable measurement and texture boundary, using a proper window size. In our experiments, we used different averaging window sizes.

3.2.2. Segmentation by Clustering in Feature Space

Features obtained through filtering approach transform the texture segmentation problem into a decision problem in the feature space. Corresponding to each pixel, there is a feature vector in which the number of features equals to the number of filters used. These feature vectors are placed in the feature space as points. If our texture features are capable of discriminating these categories then the patterns belonging to the same class will form a cluster in the feature space, which is compact and isolated from clusters corresponding to other texture categories. So each cluster represents a distinct class, and clusters of points represent different classes of patterns.

Pattern clustering algorithms are ideal vehicles for recovering such clusters in the feature space. Standard K – means algorithm is used to recover the clusters from the feature space.

3.2.2.1. K – means Algorithm

The K – means algorithm for cluster-seeking is based on the minimization of a performance index which is defined as the sum of the squared distances from all points in a cluster domain to the cluster center. The K – means algorithm requires the number of clusters to be known beforehand. This algorithm consists of the following steps:

Step 1. Determine or choose K initial cluster centers $\mathbf{z}_1(1), \mathbf{z}_2(1), \ldots, \mathbf{z}_k(1)$. Here $\mathbf{z}_1(1)$ is the texture features of the first cluster center during the first iteration.

Step 2. At the kth iteration each pixel $\mathbf{x}$ is assigned to one of the K clusters $\mathbf{C}_1(k), \mathbf{C}_2(k), \ldots, \mathbf{C}_K(k)$, where $\mathbf{C}_j(k)$ denotes the set of pixels whose cluster center is $\mathbf{z}_j(k)$. $\mathbf{x}$ is assigned to cluster $\mathbf{C}_j(k)$ if:

$$\begin{aligned} &\|\mathbf{x}-\mathbf{z}_j(k)\| < \|\mathbf{x}-\mathbf{z}_i(k)\| \\ &\forall i, j = 1,2,\ldots,K, \quad i \neq j \end{aligned} \tag{3.5}$$

Step 3. From the results of step2, compute the new cluster centers $\mathbf{z}_j(k+1)$, $j = 1, 2, \ldots, K$, such that the sum of the squared distances from all points in $\mathbf{C}_j(k)$ to the new cluster center is minimized. In other words, the new cluster center $\mathbf{z}_j(k+1)$ is computed so that the performance index

$$J_j = \sum_{\mathbf{x}\in\mathbf{C}_j(k)} \left\|\mathbf{x}-\mathbf{z}_j(k+1)^2\right\|, \quad j=1,2,\ldots,K \tag{3.6}$$

is minimized. The new cluster center which minimizes this is the sample mean of $\mathbf{C}_j(k)$. Therefore, the new cluster center is given by:

$$\mathbf{z}_j(k+1) = \frac{1}{N_j} \sum_{\mathbf{x} \in \mathbf{C}_j(k)} \mathbf{x}, \quad j = 1,2,...,K \tag{3.7}$$

where N_j is the number of pixels of the cluster $\mathbf{C}_j(k)$.

Step 4. If $\mathbf{z}_j(k+1) = \mathbf{z}_j(k)$ for $j = 1, 2, \ldots, K$, the algorithm has converged and the procedure is terminated. Otherwise, go to Step 2.

The determination of the initial cluster centers plays a crucial part because the final clustering will depend on the initial cluster centers chosen. The initial cluster centers are chosen by first generating the histograms of the feature images and then determining the dominant peaks in the histogram. An overview of the texture segmentation method is shown in Fig. 16.

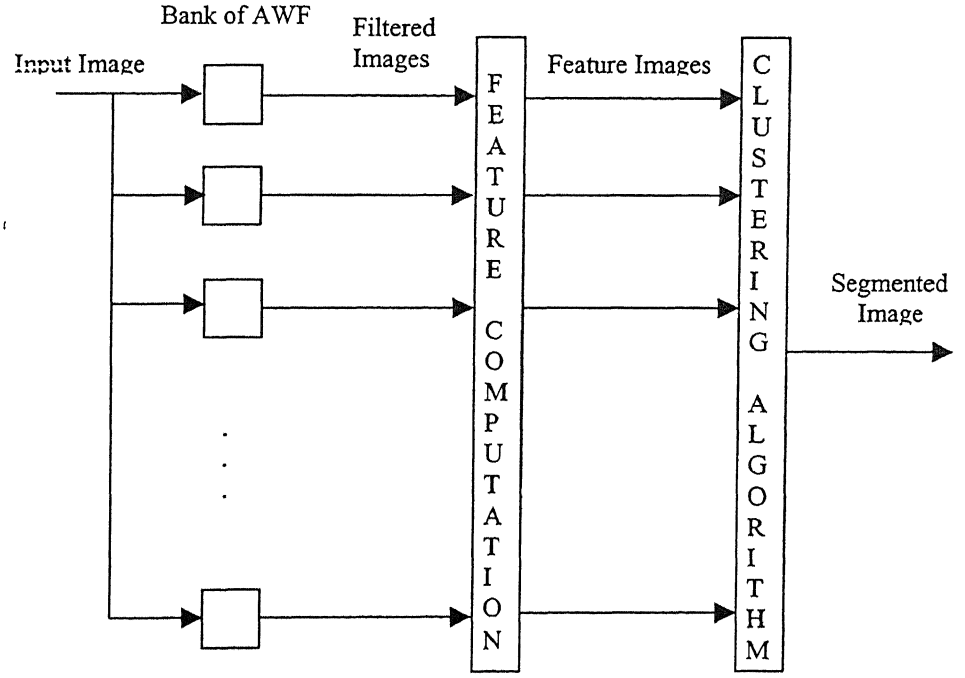


Fig. 16. An overview of the texture segmentation method

In chapter 4, the results of texture classification and segmentation, using the methods explained in this chapter, are presented.

Chapter 4

RESULTS AND DISCUSSION

In this chapter, the performance of the proposed Angular Wavelet Frames for texture classification and segmentation is presented. We have performed our experiments using three sets of filters, 'db2', 'db4' and 'db10'. For these three cases, the simulation results for texture classification on 26 Brodatz textures are given in section 4.1. Results of texture segmentation using mosaic images are presented in section 4.2.

4.1. Performance of Angular Wavelet Frames for Texture Classification

We performed classification experiments using 26 Brodatz textures. Each image taken is of size 256*256 with 256 gray levels. The sub images are obtained by filtering the texture images with 2D Angular Wavelet Frames. For each texture, feature vectors of length N, where N is the number of sub images were evaluated using Equation (3.2) with a window of size w*w over non-overlapping sub regions. For each texture, a total of (256*256)/(w*w) feature vectors were evaluated. Mean vectors and covariance matrices ($\mathbf{m}_k$, $\mathbf{C}_k$), (k = 1, 2, . . . , K), where K = 26 is the total number of classes, of the feature images are calculated. A feature vector of a given texture is assigned to a class with minimum distance value in

Equation (3.3). For each pattern tested, the training was performed on the remaining set (leaving-one-out algorithm).

The classification results using different sets of filters ('db2', 'db4' and 'db10') at different levels (level 1 – 4 sub images, level 2 – 7 sub images and level 2 – 10 sub images) with a window of size 8*8 are summarized in Table 1. For each texture a total of 1024 feature vectors are evaluated. The average classification rates of 26 textures, called the overall correct classification rate are listed in the last row of the table. From Table 1, the overall classification rate for all the three filters is poor at level 1. But there is much improvement in their performance at level 2 using 7 sub images and 10 sub images. The overall classification rate is over 94% for all the three filters. Between 'db2' and 'db4' the performance of 'db4' is better and among all the three filters, the overall classification rate is good using 'db10', which was due to the length of the filter. The classification results using windows of size 16*16 and 32*32 are given in Table 2 and Table 3 respectively. There is a significant improvement in classification rate with the windows of size 16*16 and 32*32 even at level 1. With a window of size 16*16, we have 100% correct classification rate for many images at level 2 and 100% overall classification rate using 32*32 window at level 2 with 7 and 10 sub images respectively. The results are much better when we extract features using a window of larger size. This is because as the window size increases the feature values are determined by a larger number of pixels in the window. This gives more reliable values for classification. The pattern of misclassification, that is which texture image is classified into which other category also termed as 'confusion matrix' is shown in Table 4 for the case of 'db2', level 1 – 4 sub images and using a window of size 16*16. From Table 4 we find that classification errors occurred between textures pairs that are difficult to discriminate visually. These errors are minimized with higher length filters and by extracting features using a window of larger

size. Confusion matrix for the case of 'db10' level 1 – 4 sub images, using a window of size 32*32 is given in Table 5.

A comparison of overall correct classification rates of different methods is shown in Table 6. From Table 6 we see that AWF method not only outperform DWT, which was expected due to choice of translation invariant representation but also its performance is better compared to DWF. This is because AWF can capture the dominant angular component of textures.

4.2. Simulations Results for Texture Segmentation

We have used mosaic images of two, three and four regions for segmentation task in our experiments. The reason for using mosaic images is that the texture boundaries are precisely known. But, such simple boundaries usually do not occur in real scenes. Images are filtered at level 2 by 2D angular wavelet frames constructed using 'db10'. Features are extracted from the filtered images using the Equation (3.4). These features are clustered using K – means clustering algorithm to obtain the segmentation of the images. Segmentation results are presented for different window sizes. In Fig. 17 to Fig. 20, segmentation results for two regioned texture images are shown. In Fig. 18, segmentation of the original image using a window of size 5*5 is not good, the segmentation is good with 9*9 window. In Fig. 19, segmentation using 7*7 window is not satisfactory and it is good with 11*11 window. In Fig. 21 and in Fig. 22 the segmentation results for a three regioned images are given. In Fig. 23, the segmentation results for a four regioned image are given. Using a larger window gives more reliable measurement but for accurate determination of texture boundary, the smaller window size is needed. Therefore, there is a trade-off between reliable measurement and texture boundary.

Table 1. Classification Rate using window of size 8*8

Texture Name	Level 1 - 4 sub images			Level 2 - 7 sub images			Level 2 - 10 sub images		
	db2	db4	db10	db2	db4	db10	db2	db4	db10
D11	67.38	74.02	72.66	94.92	94.43	95.41	98 24	98.93	98.63
D12	57.42	66.31	70.31	85.55	88.57	92.77	91.02	93.85	96.68
D15	71.48	78.32	82.32	89.06	90.72	93.07	91.99	94.34	96.78
D16	80.86	86.23	88.48	99.12	98.83	99.41	99.71	100	99.71
D17	89.45	95.31	93.36	98.14	99.22	99.22	99.22	99.80	99.22
D20	90.33	84.47	75.98	94.92	95.02	98.63	97.95	98.93	99.90
D21	97.95	97.95	99.12	99.22	99.71	99.90	99.61	99.80	100
D29	63.77	57.32	67.48	88.48	89.16	87.11	95.41	95.80	96.00
D34	92.77	94.82	97.46	96.00	96.00	98.54	97.17	98.24	99.80
D35	73.44	71.58	72.36	82.42	86 33	84.96	86.91	86.82	86.33
D37	91.70	91.89	91.70	97.75	97.17	97.66	97.36	98.73	98.83
D38	77.05	75.20	81.15	86.33	83.69	88.48	90 43	91.60	92.97
D49	99.71	100	100	100	100	100	100	100	100
D51	67.48	73.63	76.27	91.60	91.70	96.29	93.46	96.09	98.34
D52	65.33	73.14	74.22	98.93	99.02	99.12	99.32	99.71	99.41
D56	68.26	70.80	83.50	86.04	91.31	95.51	93.46	96.88	99.22
D66	48.83	59.57	52.73	95.61	94.14	95.70	99.32	99.51	99.80
D68	88.38	88.57	90.63	98.34	98.05	99.51	98.83	98.93	99.90
D74	72.27	78.61	88.38	94.34	96.09	96.88	97.95	99.22	98.63
D85	87.99	86.04	82.71	94.73	93.75	92.97	96.19	96.58	96.88
D87	82.42	84.18	82.32	97.85	99.02	99.71	99.51	100	99.90
D90	84.57	87.50	89.16	87.60	87.40	92.68	90.53	93.75	96.29
D101	83.40	89.06	93.36	95.41	95.12	95.90	98.05	98.73	99.12
D103	97.66	99.02	97.95	99.32	99.51	99.51	97.07	99.02	98.63
D109	87.60	84.47	89.06	97.95	97.07	98.63	98.93	99.41	99.51
D111	67.87	76.66	78.03	95.21	96.00	97.46	99.71	99.71	99.90
Overall	79.05	81.72	83.49	94.03	94.50	95.96	96.44	97.48	98.09

Table 2. Classification rate using window of size 16*16

Texture Name	Level 1 - 4 sub images			Level 2 - 7 sub images			Level 2 - 10 sub images		
	db2	db4	db10	db2	db4	db10	db2	db4	db10
D11	92.58	94.92	96.48	100	99.61	99.22	100	100	100
D12	82.42	83.59	90.23	98.05	97.66	99.22	98 44	100	99.61
D15	82.81	87.89	91.41	96.88	97.27	98.44	98.05	98.44	99.61
D16	91.80	94.14	98.05	100	100	100	100	100	100
D17	98.83	99.61	99.61	100	100	100	100	100	100
D20	98.44	97.66	93.36	99.61	100	100	100	100	100
D21	100	100	100	100	100	100	100	100	100
D29	77.73	79.30	84.77	100	98.44	98.83	100	100	100
D34	98.83	99.22	100	99.61	98.83	100	100	100	100
D35	87.11	92.19	93.36	94.92	96.48	96.88	98.05	98.44	97.27
D37	96.09	96.48	95.31	100	98.44	100	99.22	100	100
D38	87.50	85.55	94.14	98.44	97.66	98.83	96.88	97.66	100
D49	100	100	100	100	100	100	100	100	100
D51	83.98	87.11	91.41	97.27	99.22	99.61	99.22	99.61	100
D52	85.55	91.02	90.23	100	100	100	100	100	100
D56	98.44	96.88	99.22	100	99.22	99.61	100	100	100
D66	72.27	88.28	79.69	100	100	100	100	100	100
D68	94.53	94.92	96.48	100	99.61	100	100	100	100
D74	98.05	98.83	99.22	100	100	100	100	100	100
D85	96.88	98.05	97.66	99.61	99.61	96.09	100	100	98.83
D87	98.05	97.66	98.44	100	100	100	100	100	100
D90	94.92	96.09	96.09	97.27	98.83	99.22	99.22	99.22	100
D101	99.61	99.61	99.61	100	100	100	100	100	100
D103	100	100	100	100	100	100	100	100	100
D109	93.75	92.58	97.27	99.61	100	99.61	100	100	100
D111	94.53	97.27	97.27	100	100	100	100	100	100
Overall	92.49	94.19	95.36	99.28	99.26	99.44	99.58	99.74	99.82

Table 3. Classification rate using window of size 32*32

Texture Name	Level 1 - 4 sub images			Level 2 - 7 sub images			Level 2 - 10 sub images		
	db2	db4	db10	db2	db4	db10	db2	db4	db10
D11	100	100	100	100	100	100	100	100	100
D12	93.75	98.44	98.44	100	100	100	100	100	100
D15	87.50	90.63	100	100	100	100	100	100	100
D16	98.44	96.88	100	100	100	100	100	100	100
D17	100	100	100	100	100	100	100	100	100
D20	100	100	98.44	100	100	100	100	100	100
D21	100	100	100	100	100	100	100	100	100
D29	93.75	90.63	93.75	100	100	100	100	100	100
D34	100	100	100	100	100	100	100	100	100
D35	96.88	100	100	100	100	100	100	100	100
D37	100	100	96.88	100	100	100	100	100	100
D38	98.44	98.44	98.44	100	100	100	100	100	100
D49	100	100	100	100	100	100	100	100	100
D51	95.31	96.88	98.44	100	100	100	100	100	100
D52	98.44	98.44	98.44	100	100	100	100	100	100
D56	100	100	100	100	100	100	100	100	100
D66	95.31	100	89.06	100	100	100	100	100	100
D68	98.44	98.44	100	100	100	100	100	100	100
D74	100	100	100	100	100	100	100	100	100
D85	100	100	100	100	100	100	100	100	100
D87	100	100	100	100	100	100	100	100	100
D90	100	100	98.44	100	100	100	100	100	100
D101	100	100	100	100	100	100	100	100	100
D103	100	100	100	100	100	100	100	100	100
D109	96.88	98.44	100	100	100	100	100	100	100
D111	98.44	100	100	100	100	100	100	100	100
Overall	98.14	98.74	98.86	100	100	100	100	100	100

Table 4. Confusion matrix for the case of 'db2', level 1 using window of size 16*16

	D11	D12	D15	D16	D17	D20	D21	D29	D34	D35	D37	D38	D49	D51	D52	D56	D66	D68	D74	D85	D87	D90	D101	D103	D109	D111	% rate
D11	237	3	0	0	2	0	0	1	0	3	0	0	0	1	0	0	2	0	2	0	0	0	0	0	0	5	92.58
D12	3	211	27	0	0	0	0	0	0	4	0	0	0	4	0	0	0	0	0	0	2	2	0	0	0	3	82.42
D15	0	34	212	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	1	0	7	0	0	0	0	0	82.81
D16	2	0	0	235	0	0	0	16	0	0	0	0	0	0	2	0	1	0	0	0	0	0	0	0	0	0	91.80
D17	0	0	0	0	253	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	1	98.83
D20	0	0	0	0	0	252	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	98.44
D21	0	0	0	0	0	0	256	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100.0
D29	7	0	0	2	0	0	0	199	0	0	0	0	0	0	27	0	18	0	0	0	0	0	0	0	0	3	77.73
D34	0	0	0	0	0	0	0	0	253	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	98.83
D35	6	26	0	0	0	0	0	0	0	223	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	87.11
D37	0	0	0	0	0	0	0	0	0	0	246	0	0	0	0	0	1	0	0	0	0	9	0	0	0	0	96.09
D38	2	1	0	1	0	0	0	3	0	0	1	224	0	1	0	3	0	0	0	0	0	20	0	0	0	0	87.50
D49	0	0	0	0	0	0	0	0	0	0	0	0	256	0	0	0	0	0	0	0	0	0	0	0	0	0	100.0
D51	4	21	2	0	0	0	0	0	0	1	0	0	0	215	0	0	0	11	2	0	0	0	0	0	0	0	83.98
D52	0	0	0	0	0	0	0	5	0	0	0	0	0	0	219	0	11	0	0	0	0	1	0	0	20	0	85.55
D56	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	252	0	0	2	0	0	0	0	0	0	2	98.44
D66	0	7	0	0	0	0	0	6	0	0	0	0	0	0	39	0	185	0	0	0	0	0	0	0	4	15	72.27
D68	0	3	0	0	0	0	0	0	0	0	0	0	0	11	0	0	0	242	0	0	0	0	0	0	0	0	94.53
D74	0	3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	251	0	0	1	0	0	0	0	98.05
D85	2	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	5	248	0	0	0	0	0	0	96.88
D87	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	251	0	0	0	0	0	98.05
D90	0	0	0	0	0	0	0	0	0	0	0	3	0	0	2	3	2	0	3	0	0	243	0	0	0	0	94.92
D101	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	255	0	0	0	99.61
D103	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	256	0	0	100.0
D109	1	2	0	0	0	0	0	0	0	0	0	0	0	0	7	0	5	0	0	0	0	0	0	0	240	1	93.75
D111	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	1	0	3	0	0	0	0	242	94.53

Overall Classification Rate = 92.49%

Table 5. Confusion matrix for the case of 'db10', level 1 using window of size 32*32

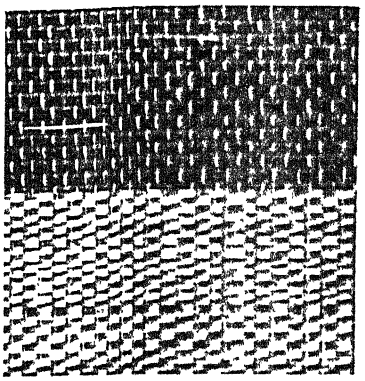
	D11	D12	D15	D16	D17	D20	D21	D29	D34	D35	D37	D38	D49	D51	D52	D56	D66	D68	D74	D85	D87	D90	D101	D103	D109	D111	% rate
D11	64	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
D12	0	63	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	98.4
D15	0	0	64	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
D16	0	0	0	64	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
D17	0	0	0	0	64	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
D20	0	1	0	0	0	63	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	98.4
D21	0	0	0	0	0	0	64	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
D29	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	93.8
D34	0	0	0	0	0	0	0	0	64	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
D35	0	0	0	0	0	0	0	0	0	64	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
D37	0	0	0	0	0	0	0	0	0	0	62	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	96.9
D38	0	0	0	0	0	0	0	0	0	0	0	63	0	0	0	0	0	0	0	0	0	1	0	0	0	0	98.4
D49	0	0	0	0	0	0	0	0	0	0	0	0	64	0	0	0	0	0	0	0	0	0	0	0	0	0	100
D51	0	1	0	0	0	0	0	0	0	0	0	0	0	63	0	0	0	0	0	0	0	0	0	0	0	0	98.4
D52	0	0	0	0	0	0	0	0	0	0	0	0	0	0	63	0	1	0	0	0	0	0	0	0	0	0	98.4
D56	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	64	0	0	0	0	0	0	0	0	0	0	100
D66	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7	0	57	0	0	0	0	0	0	0	0	0	100
D68	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	64	0	0	0	0	0	0	0	0	100
D74	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	64	0	0	0	0	0	0	0	89.1
D85	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	64	0	0	0	0	0	0	100
D87	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	64	0	0	0	0	0	100
D90	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	63	0	0	0	0	100
D101	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	64	0	0	0	100
D103	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	64	0	0	100
D109	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	64	0	98.4
D111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	64	100

Overall Classification Rate = 98.86%

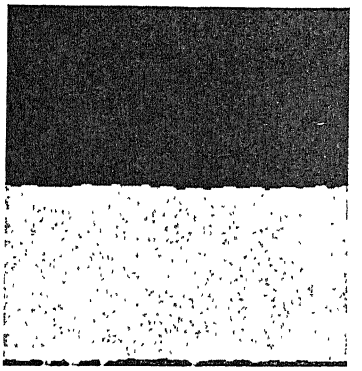
Table 6. Comparison of Overall Correct Classification Rates (window size = 32*32)

Method	Classification Rate	
	Level 1 – 4 sub images	Level 2 – 7 sub images
DWT	91.93%	95.44%
DWF	96.48%	98.83%
AWF	**98.14%**	**100%**

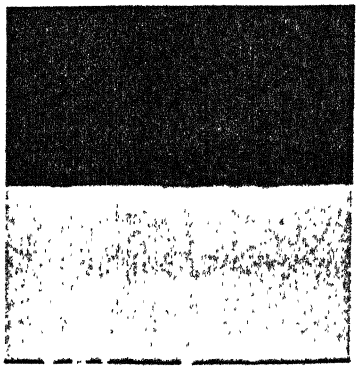
Note: Results printed in light letter is directly taken from Reference [10] for comparison



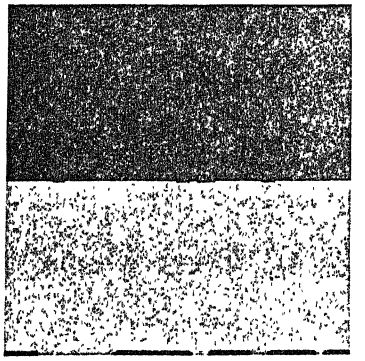
(a)



(b)

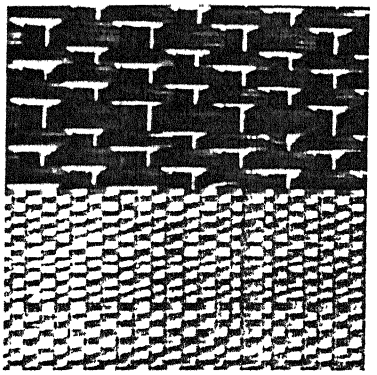


(c)



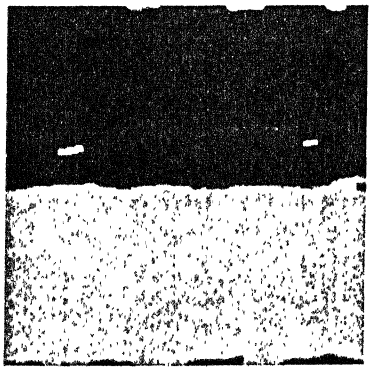
(d)

Figure. 17. (a) Original Image. Segmentation results using a window of size (b) 5*5 (c) 7*7 (d) 9*9

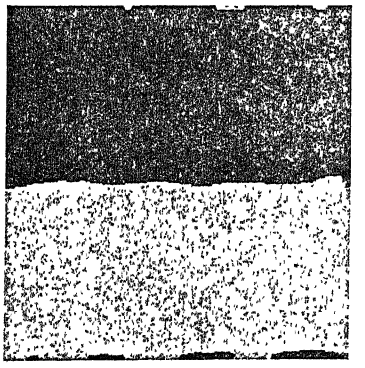


(a)

(b)



(c)



(d)

Figure. 18. (a) Original Image. Segmentation results using a window of size (b) 5*5 (c) 7*7 (d) 9*9

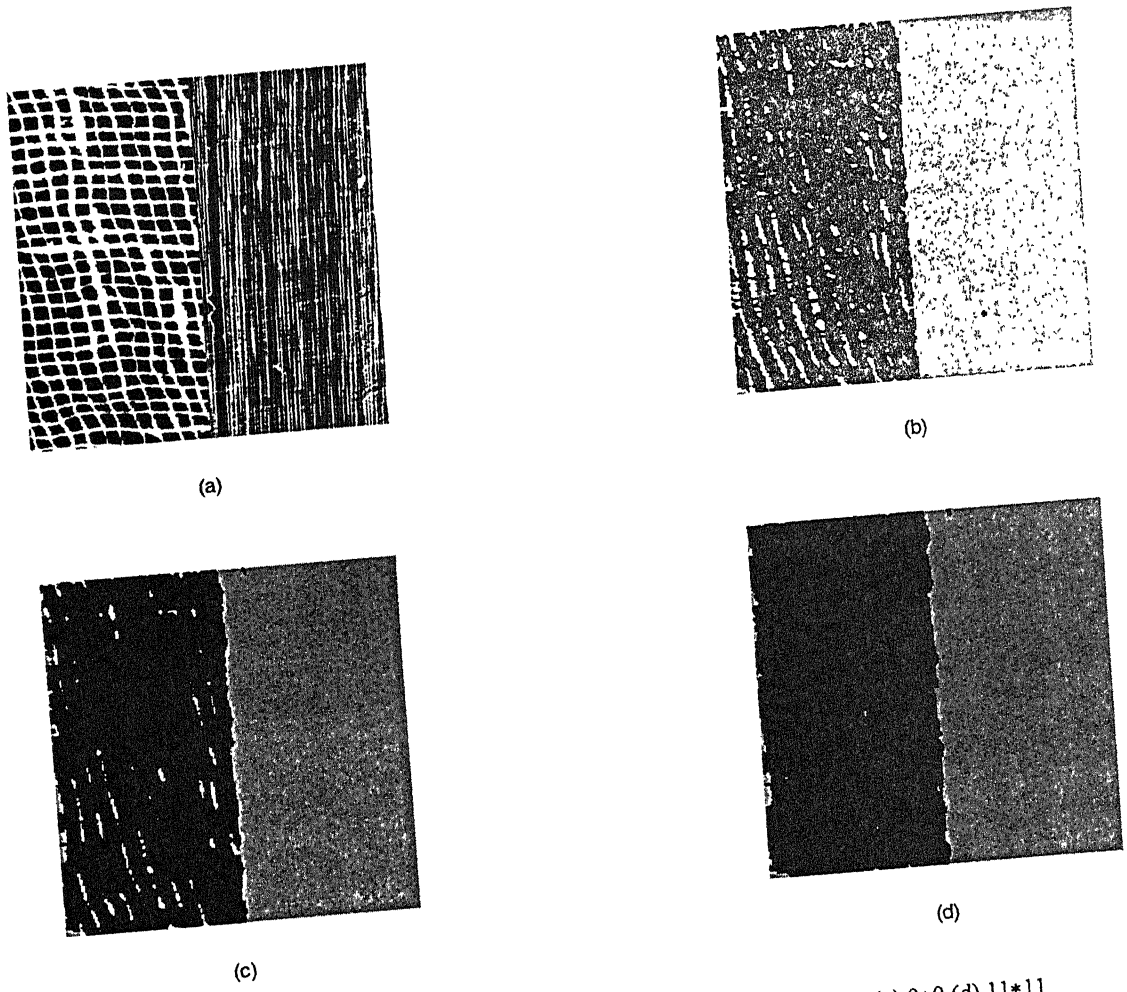
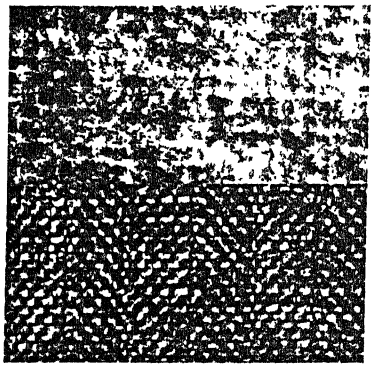
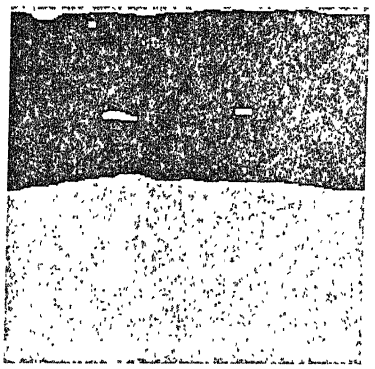


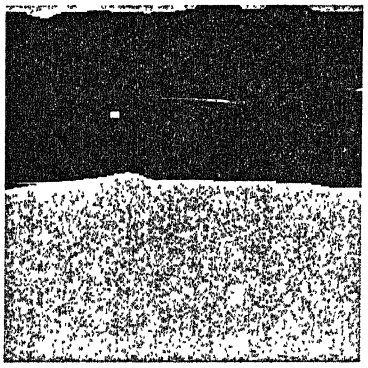
Fig. 19. (a) Original Image. Segmented using a window of size (b) 7*7 (c) 9*9 (d) 11*11



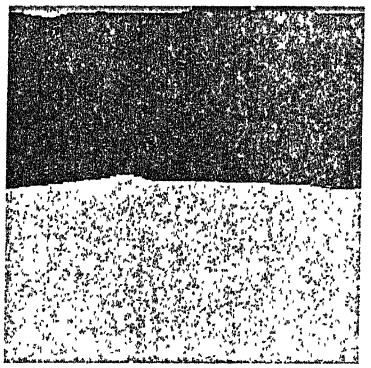
(a)



(b)



(c)



(d)

Fig. 20. (a) Original image. Segmentation results using window of size (b) 7*7 (c) 9*9 (d) 11*11

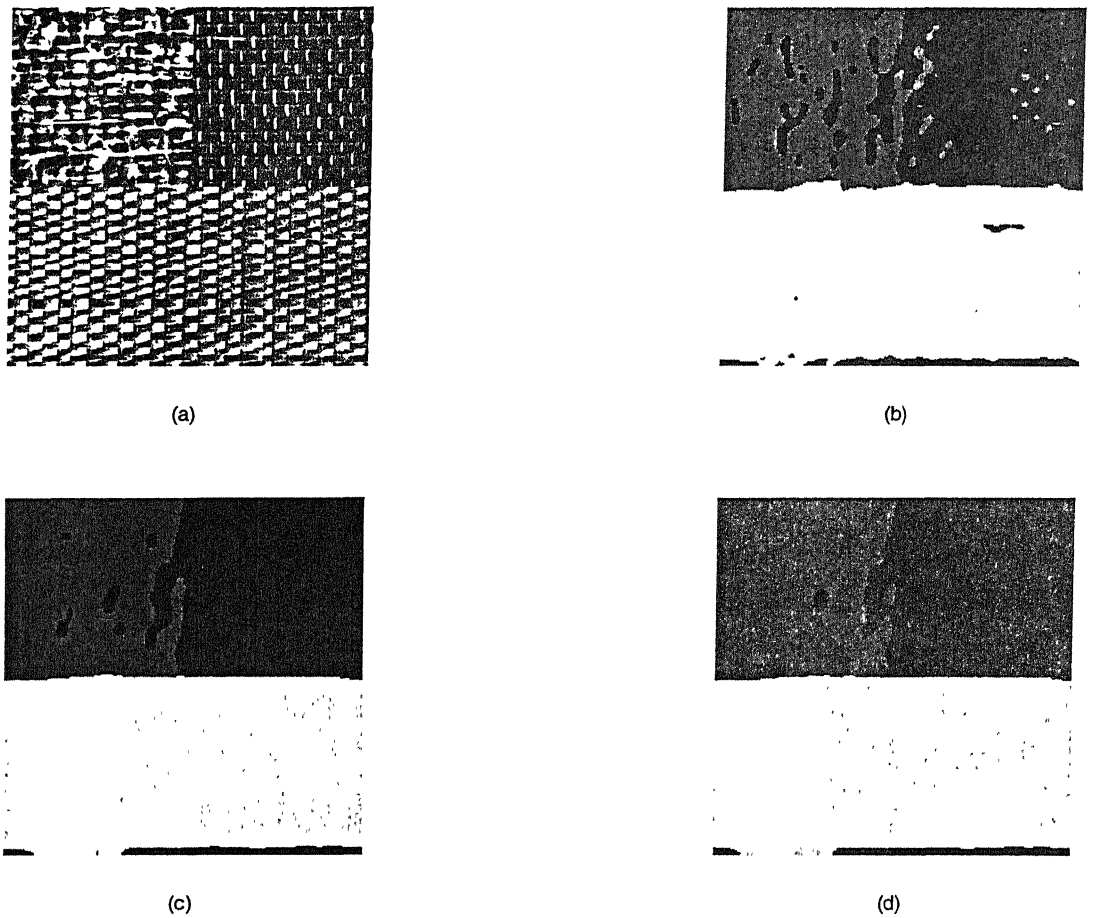


Fig. 21. (a) Original Image. Segmentation results using window of size (b) 7*7 (c) 11*11 (d) 13*13

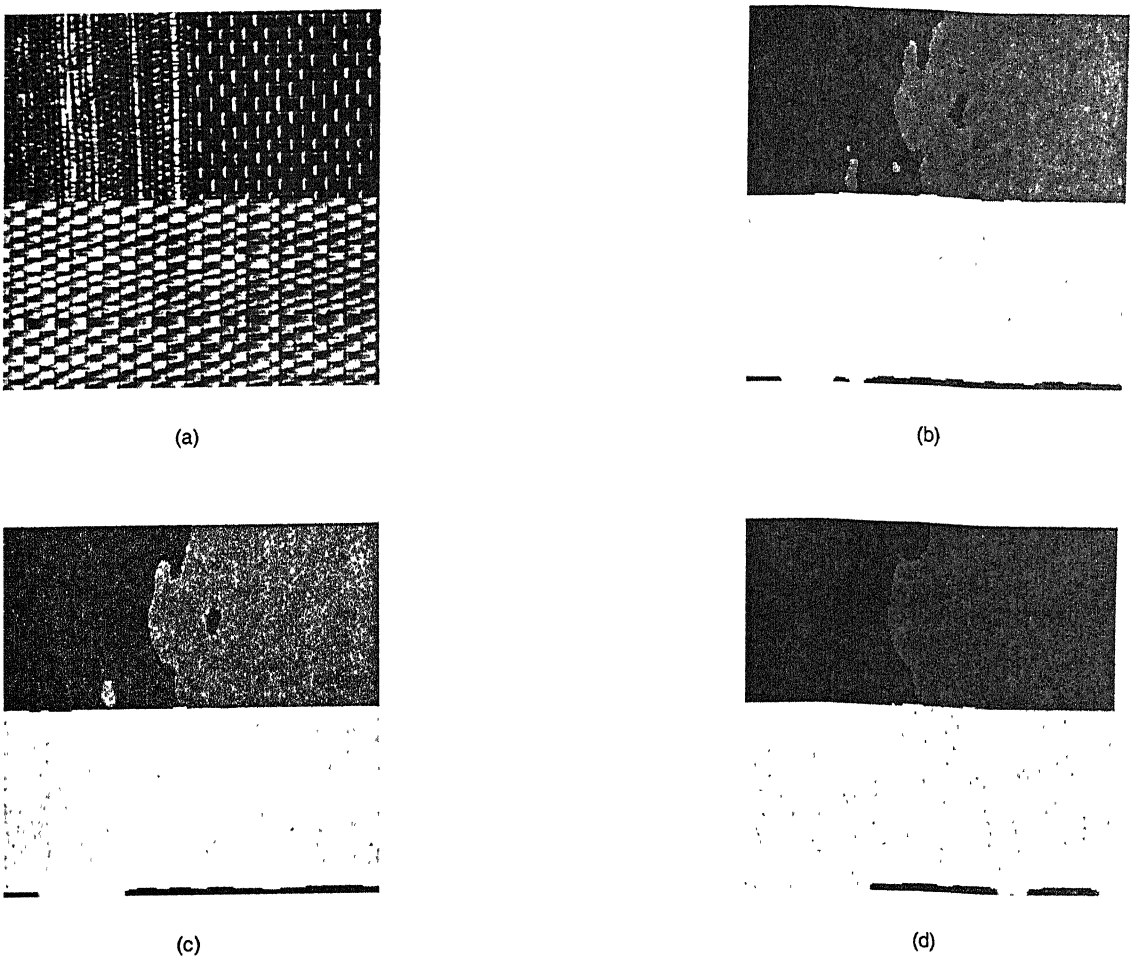
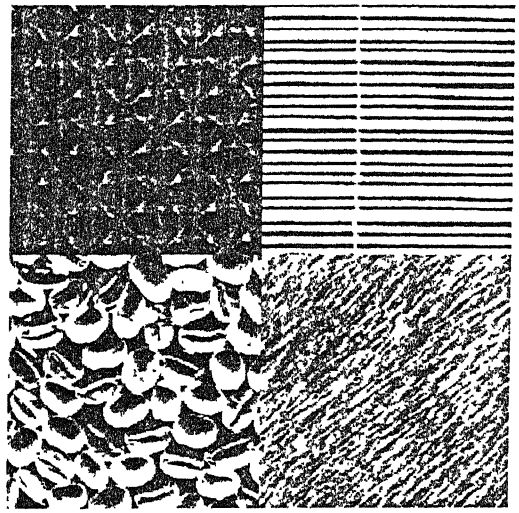
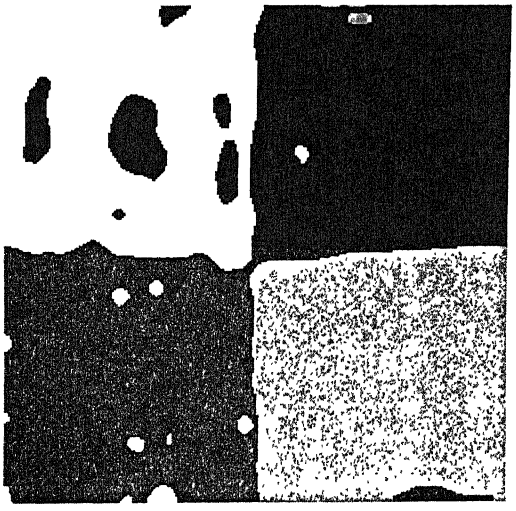


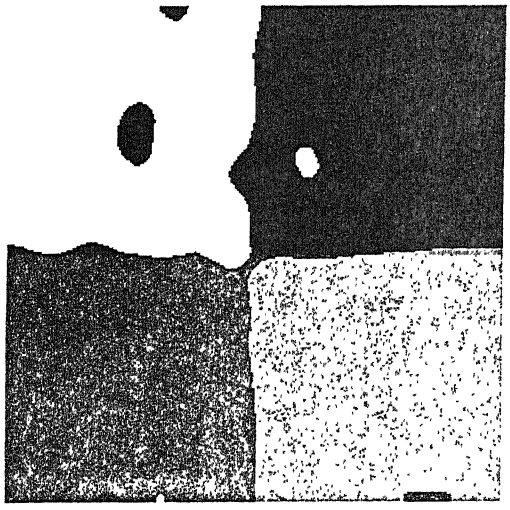
Fig. 22. (a) Original Image. Segmented using a window of size (b) 9*9 (c) 11*11 (d) 15*15



(a)



(b)



(c)

Fig. 23. (a) Original Image. Segmented Image using a window of size (b) 11*11 (c) 17*17

Chapter 5

CONCLUSIONS AND FUTURE SCOPE

5.1. CONCLUSIONS

We proposed the method of constructing 2D Angular Wavelet Frames using circular and angular transformations. We have proved that the constructed 2D filters using the transformations of 1D PRFB, along with their conjugates at the synthesis end constitute a PRFB. Our classification results demonstrate that AWF approach is a promising approach for texture characterization. We observe from Table 5, that the rate of misclassification is less for textures having visually dominant angular component. It suggests that the constructed AWF is well suited for textures having dominant angular component.

The segmentation results show that the 2D AWF is capable of segmenting multi-texture image. The segmentation results are good for two and three regioned texture images. The regions of the texture image segmentation are uniform and there is no hole in the interior regions. For three and four regioned images, the boundary is not spatially accurate and for four regioned image there are holes in the interior regions. The results could be improved by using spatial relationship among the pixels. The segmentation of the feature images has been realized by

applying k – means clustering method, where the number of regions are known before, and the initial cluster centers are to be chosen properly so that they lie close to the clusters in the feature space. So the method proposed is supervised.

5.2. FUTURE SCOPE

The proposed 2D AWF method can be modified to classify Rotation Invariant Textures by combining the filters for a given level and for all angles. This can also be used to classify Scale Invariant Textures by combining the filters for all the levels given an angle (orientation). Segmentation can be made unsupervised by using merging techniques or by modifying the standard k – means algorithm so that it can iteratively determine the number of clusters in the feature space. For a large number of regions in an image, the boundaries were not spatially accurate. This could be overcome by using some spatial relationship among the pixels.

REFERENCES

[1] R. M Haralick, K. Shanmugan, and I. Dinstein, "Textural features for image classification," *IEEE Trans. Sys., Man, Cybern.*, vol. SMC-8, no.6, pp. 610-621, Nov. 1973.

[2] I. Daubechies, "Orthogonal bases of compactly supported wavelets," *Comm. Pure Appl. Math.*, vol. 41, pp. 909-996, 1988.

[3] S. G. Mallat, "A theory of multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 11, no. 7, pp. 674-693, 1989.

[4] O. Rioul and M. Vetterli, "Wavelets and signal processing," *IEEE Signal Processing Mag.*, vol. 8, no. 4, pp. 11-38, Oct. 1991.

[5] A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using Gabor filters," *Pattern Recognition*, vol. 24, no. 12, pp. 1167-1186, 1991.

[6] A. C. Bovik, M. Clark, and W. S. Geisler, "Multichannel texture analysis using localized spatial filters," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 12, no. 1, pp. 55-73, Jan. 1990.

[7] T. Chang and C. C. J. Kuo, "Texture analysis and classification with tree-structured wavelet transform." *IEEE Trans. Image Processing*, vol. 2, no. 4, pp. 429-441, Oct. 1993.

[8] D. Dunn, W. E. Higgins, and Joseph Wakeley, "Texture segmentation using 2D Gabor elementary functions," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 16, no. 2, pp. 130-149, Feb. 1994.

[9] D. Dunn, and W. E. Higgins, "Optimal Gabor filters for texture segmentation," *IEEE Trans. Image Processing*, vol. 4, no. 7, pp. 947-964, July 1995.

[10] M. Unser, "Texture classification and segmentation using wavelet frames," *IEEE Trans. Image Processing*, vol. 4, no. 11, pp. 1549-1560, Nov. 1995.

[11] E. Salari, and Z. Ling, "Texture segmentation using hierarchical wavelet decomposition," *Pattern Recognition*, vol. 28, no. 12, pp. 1819-1824, 1995.

[12] D. P. Mittal, "Texture segmentation using Gabor filters," *4th Int. Conf. On Knowledge-Based Intelligent Eng. Systems & Allied Technologies*, pp. 109-112, 30th Aug.-1st sept. 2000, Brighton, UK.

[13] His-Chin Hsin, "Texture segmentation using modulated wavelet transform," *IEEE Trans. Image Processing*, vol. 9, no. 7, pp. 1299-1302, July 2000.

[14] O. Rioul, "A discrete-time multiresolution theory," *IEEE Trans. Signal Processing*, vol. 41, no. 8, pp. 2591-2606, Aug. 1993.

[15] Rajnish Prasad, "Texture classification using Angular Wavelet Frames," *B. Tech. Project Report*, 2000, Dept. of EE, IIT – Kanpur.